

**UNIVERSIDAD AUTONOMA DE MADRID**

**ESCUELA POLITECNICA SUPERIOR**



**Grado en Ingeniería Informática**

## **TRABAJO FIN DE GRADO**

**ESTIMULANDO LA CONDUCCIÓN EFICIENTE  
MEDIANTE UNA APP PARA ANDROID**

**Virginia Martín Burgos**  
**Tutor: Francisco Jurado Monroy**  
**Ponente: Jaime Moreno Llorena**

**JULIO 2018**



# **ESTIMULANDO LA CONDUCCIÓN EFICIENTE MEDIANTE UNA APP PARA ANDROID**

**AUTORA: Virginia Martín Burgos**

**TUTOR: Francisco Jurado Monroy**

**Dpto. de Ingeniería Informática  
Escuela Politécnica Superior  
Universidad Autónoma de Madrid  
Julio de 2018**



## Resumen (castellano)

Hoy en día, se intenta concienciar a la sociedad de que hay que tener una buena conducción para la reducción de la polución. Este tema es uno de los más preocupantes en la sociedad actual, ya que hay ciertas ciudades en las que la contaminación está originando un problema realmente serio. El medio ambiente no podrá hacer nada solo, es el ser humano el que tiene que intentar reducir ese factor. Los seres humanos pueden contribuir reduciendo la velocidad de los vehículos y teniendo una conducción eficiente.

Además, cada día hay miles de accidentes en las carreteras, la DGT realiza campañas constantemente para lograr que los usuarios se conciencien de la importancia de conducir de forma segura, de manera que se reduzca la siniestralidad en las carreteras. Estos accidentes se podrían prevenir si los usuarios llevasen una buena conducción y fuesen prudentes al volante.

Para intentar ayudar a los usuarios a que lleven ese tipo de conducción, se realizará una aplicación móvil, la cual llamaremos *SafeCar*. En ella el usuario podrá introducir el coche con el que suele conducir. Tras ello, podrá comenzar a realizar trayectos, los cuales se evaluarán mediante su velocidad. De esta forma, el usuario obtendrá una puntuación para cada trayecto. Después el usuario podrá realizar un análisis de esos trayectos, observando gráficas de los mismos. Además, podrá comparar unos trayectos con otros. De esta manera, el usuario podrá ver su progresión, es decir, cómo ha ido mejorando o empeorando en sus trayectos, pudiendo así valorar su propia conducción de forma que pueda tomar decisiones para los futuros trayectos que realice.

Por lo tanto, la aplicación se presenta como un reto para el usuario, de manera que este pueda ayudar no solo al medio ambiente sino también al resto de usuarios de las vías públicas, reduciendo lo máximo posible los accidentes.

## Palabras clave (castellano)

Conducción eficiente, GPS, Android, dispositivos móviles, multas, carreteras, coches, trayectos, análisis, conducción.

## Abstract (English)

Nowadays, we try to make society aware of the need of having a good driving to reduce pollution. This issue is one of the most worrying ones nowadays since in certain cities pollution is taking over it. The environment can not do anything alone, it is the human being who has to try to reduce that factor. Human beings can contribute by reducing the speed of vehicles and having an efficient driving.

In addition, every day there are thousands of accidents on the roads, the DGT constantly carries out campaigns to make users aware of the importance of reducing them. That could be prevented if the users had a good driving and they were cautious behind the wheel.

We can try to help users to carry that type of driving with a mobile application that we will make, which we will call *SafeCar*. In it the user can introduce the car he usually drives. After that, you can begin to make trips, which will be evaluated by their speed and thus be able to give the user a score. Then the user can perform an analysis of these paths observing graphs. In addition, you can compare some journeys with others, which are labelled with the same name. In that way, the user will be able to see his progression, that is, how he has been improving or getting worse in his journeys so that he can evaluate his driving and then he would be able to make decisions for future journeys.

The purpose of the application is to give a challenge to the user, and with this help not only the environment but also roads to reduce accidents as much as possible.

## Keywords (inglés)

Efficient driving, GPS, Android, mobile devices, fines, roads, cars, routes, analysis, driving.

## *Agradecimientos*

En primer lugar, quiero dar las gracias a mis padres, ya que sin ellos no podría estar escribiendo ahora mismo. Por la confianza que han depositado en mí durante estos cuatro años, que la verdad es que no han sido nada fáciles. Han tenido que soportar mis lágrimas, mis gritos y sobre todo las violaciones de segmento que han ido apareciendo, que han sido más bien muchas. Sin ellos, yo no estaría aquí, han sido mi pilar tanto económico como psicológico. También, a mi hermana, que ha sido a la persona que más he echado de menos en estos años, ya que siempre hemos estado acostumbradas a estar juntas y el cambio de un pueblo a Madrid ha sido muy grande. Pero no ha importado, siempre ha estado ahí para darme apoyos y ánimos y cada fin de semana darme los abrazos suficientes para toda la semana. Y por último, no puedo olvidar a mis abuelos, que siempre han estado pendientes de mí, y han apoyado mis decisiones, tanto las buenas como las malas.

También, dar las gracias, a mis compañeros de clase, los cuales me han ayudado en el transcurso de la carrera. Y sobre todo dar gracias, a mi grupo, “Yatekomo”, con el cuál he pasado muchos momentos buenos y malos. Pero sin duda, lo mejor ha sido que he encontrado a mi mejor amiga, la que me ha aguantado en todas las prácticas de la carrera. Pero no todo ha sido malo, también he disfrutado con ella como con nadie, y las prácticas han llegado a ser hasta divertidas.

Para terminar, dar gracias a los profesores, a todo el equipo docente y sobre todo a mi tutor que ha estado ahí, conmigo, resolviendo todas y cada una de las dudas que me han ido surgiendo y ayudándome a resolverlas.





## INDICE DE CONTENIDOS

1	Introducción.....	1
1.1	Motivación.....	1
1.2	Objetivos.....	1
1.2.1	Requisitos funcionales.....	2
1.2.2	Requisitos no funcionales.....	2
1.3	Organización de la memoria.....	3
2	Estado del arte.....	5
2.1	Conducción eficiente.....	5
2.2	Aplicaciones similares y recursos útiles.....	6
2.3	Elección del sistema operativo Android.....	7
2.4	Comentarios finales.....	9
3	Metodología y herramientas utilizadas.....	11
3.1	Metodología usada.....	11
3.2	Plataformas y tecnologías.....	13
3.3	Sistema operativo: Android.....	14
4	Análisis y Diseño.....	15
4.1	Introducción.....	15
4.2	Casos de uso.....	15
4.2.1	Caso de uso de registro.....	16
4.2.2	Caso de uso de inicio de sesión.....	16
4.2.3	Caso de uso de cambiar opciones de perfil.....	16
4.2.4	Caso de uso de introducción de un coche.....	16
4.2.5	Caso de uso de introducción de trayecto.....	17
4.2.6	Caso de uso de ver los trayectos.....	17
4.2.7	Caso de uso de análisis del trayecto.....	17
4.2.8	Caso de uso de la visualización de la clasificación.....	18
4.3	Diagrama de clases.....	19
4.4	Maquetas de la interfaz de usuario.....	19
4.5	Diseño de la base de datos.....	20
5	Desarrollo.....	25
5.1	Introducción.....	25
5.2	Patrón de diseño utilizado.....	25
5.3	Cuestiones y detalles de implementación de la base de datos.....	26
5.4	Cuestiones y detalles de implementación del sistema GPS.....	27
5.5	Interfaz de usuario final.....	28
5.5.1	Guía de usuario.....	29
5.5.2	Comparación con la maqueta.....	33
5.6	Lectura de las gráficas.....	34
5.6.1	Estudio de las gráficas.....	34
5.6.2	Ejemplo de lectura.....	35
6	Pruebas y resultados.....	37
6.1	Introducción.....	37
6.2	Pruebas unitarias.....	37
6.3	Pruebas de usabilidad.....	37
6.4	Prueba de validación.....	40
6.5	Pruebas de verificación.....	40

7 Conclusiones y trabajo futuro.....	41
7.1 Conclusiones.....	41
7.1.1 Conclusiones de objetivos .....	41
7.1.2 Conclusiones personales.....	42
7.2 Trabajo futuro .....	42
Referencias .....	43
Glosario .....	45
Anexos.....	I
A Maqueta.....	I
B Plantilla del Sistema de Usabilidad (SUS).....	V
C Pruebas de validación.....	VII
C.1 Prueba para el caso de uso “Introducir coche” .....	VII
C.2 Prueba para el caso de uso “Introducir trayecto” .....	X

## INDICE DE FIGURAS

FIGURA 2-1: GRÁFICAS ANDROID O IOS REALIZADAS EN EEUU .....	8
FIGURA 2-2: GRÁFICA ANDROID O IOS MUNDIAL.....	8
FIGURA 2-3: GRÁFICA ANDROID O IOS EN TABLETS .....	9
FIGURA 3-1: EJEMPLO DE PLANIFICACIÓN .....	11
FIGURA 3-2: DIAGRAMA DURACIÓN FASES DE LA METODOLOGÍA EMPLEADA .....	12
FIGURA 4-1: DIAGRAMA DE CASOS DE USO.....	15
FIGURA 4-2: DIAGRAMA DE CLASES SIMPLIFICADO .....	19
FIGURA 4-3: MODELO ENTIDAD-RELACIÓN .....	20
FIGURA 5-1: RESUMEN MVC .....	25
FIGURA 5-2: CAPTURAS DE LA INTERFAZ DEL USUARIO .....	31
FIGURA 5-3: GRÁFICA DE TRAYECTO .....	35
FIGURA 6-1: GRÁFICA DE PUNTUACIÓN DE LOS USUARIOS .....	38
FIGURA A-1: MAQUETA DE LA APLICACIÓN .....	II
FIGURA C-1: TRANSICIÓN A LAS OPCIONES DE LA APLICACIÓN .....	VIII
FIGURA C-2: TRANSICIÓN PARA INTRODUCIR UN COCHE .....	VIII
FIGURA C-3: TRANSICIÓN A LA PANTALLA PRINCIPAL CON TRAYECTOS .....	IX
FIGURA C-4: INTRODUCIR UN COCHE.....	IX

FIGURA C-5: ERROR AL INTRODUCIR UN COCHE.....	X
FIGURA C-6: TRANSICIÓN A LAS OPCIONES DE LA APLICACIÓN .....	XI
FIGURA C-7: TRANSICIÓN PARA INTRODUCIR UN TRAYECTO .....	XI
FIGURA C-8: INTRODUCIR UNA RUTA.....	XII
FIGURA C-9: TRANSICIÓN DE LA FINALIZACIÓN DE UN TRAYECTO A LA PANTALLA PRINCIPAL ....	XII

## INDICE DE TABLAS

TABLA 1 : PUNTOS POR INFRACCIONES COMETIDAS .....	18
TABLA 2 : PUNTOS Y SANCIÓN POR VELOCIDADES .....	23



# 1 Introducción

---

## 1.1 Motivación

Escoger una propuesta para este proyecto no es tarea fácil, hay un gran abanico de posibilidades donde elegir. Uno de los temas que más me apasionan son los coches, por lo que tener la posibilidad de realizar un proyecto sobre este aspecto es para mí una gran motivación.

Lo que se ha planteado en el presente Trabajo Fin de Grado (TFG) ha sido la realización de una aplicación Android para fomentar la conducción eficiente, de tal manera que un conductor pueda percibir cómo y de qué manera ha realizado sus trayectos; y lo más importante, aprender para futuras ocasiones y así poder lograr una conducción adecuada.

Este tema hoy en día está muy en auge debido a que a diario hay accidentes por errores que se realizan en la conducción, y la mayoría debidos al exceso de velocidad. Por ello hay que tomar conciencia e intentar aportar todo lo posible a la sociedad para que los accidentes disminuyan.

En la actualidad, cada día emergen nuevas tecnologías enfocadas a la seguridad en las carreteras y a la conducción eficiente; existen también multitud de campañas para concienciar a los conductores acerca de la importancia de respetar los límites de velocidad y de conducir de manera prudente, y a veces estas no son la mejor manera de motivar a los usuarios de las vías públicas a ser precavidos al volante. Si un conductor pudiese ver su progreso y ver que el esfuerzo tiene una recompensa, es decir, obtiene una mayor puntuación tras realizar un trayecto, y además es el primero de su círculo en el ranking, puede que le motive a llevar una mejor conducción.

## 1.2 Objetivos

El objetivo de este proyecto, es diseñar e implementar una aplicación en la cual se pueda informar a los conductores acerca de los errores que comenten en sus trayectos por carretera para que posteriormente puedan, en base a la información obtenida por la aplicación, mejorar su forma de conducir. Para ello, crearemos la aplicación *SafeCar*, que podrá ser descargada por cualquier conductor y desde la cual se podrán realizar trayectos de forma que la aplicación recopile información sobre los mismos y se la presente al usuario, pudiendo este mejorar la forma en la que conduce en base a ello.

Para alcanzar el objetivo planteado, se definen los siguientes requisitos (en el apartado de casos de uso se explicarán con más detalle).

### **1.2.1 Requisitos funcionales**

**RF-1.-**El usuario tendrá la posibilidad de registrarse la primera vez que se descargue la aplicación.

**RF-2.-**El usuario una vez registrado podrá utilizar la aplicación iniciando sesión introduciendo su e-mail y su contraseña previamente establecidos en su registro.

**RF-3.-**Si el usuario ha olvidado la contraseña, podrá tener la opción de recuperarla mediante el envío de una nueva contraseña al correo o a la plataforma que especifique el usuario.

**RF-4.-**El usuario podrá introducir diferentes coches con los que realizar los diferentes trayectos.

**RF-5.-**El usuario podrá introducir trayectos en la aplicación (es decir, crear un nuevo trayecto a partir de los datos de su ubicación), los cuales se guardarán en la base de datos para su posterior consulta.

**RF-6.-**Una vez realizado un trayecto, el usuario podrá consultarlo y analizar en qué puntos del trayecto ha realizado una conducción de acuerdo a las normas de tráfico o dónde, por el contrario, ha cometido alguna infracción que le acarree algún tipo de sanción económica o penalización.

**RF-7.-**El usuario podrá comparar varios trayectos si así lo desea, siempre que en el momento de su creación se hayan introducido con el mismo nombre. Cuando un trayecto se haya realizado más de una vez, se podrá ver la progresión de cómo se ha ido realizando, para ver si se ha empeorado o mejorado con respecto a trayectos anteriores registrados por la aplicación.

**RF-8.-** Se dispondrá de una clasificación en la que el usuario podrá consultar su posición y comprobar así como de buena es su conducción con respecto al resto de personas que utilizan la aplicación.

### **1.2.2 Requisitos no funcionales**

**RNF-1.-** La aplicación será compatible con las API's de Android.

**RNF-2.-** El teléfono que se vaya a utilizar debe incorporar sistema GPS.

**RNF-3.-** Será imprescindible que el teléfono móvil disponga de Internet.

**RNF-4.-** El sistema será usable de acuerdo con las métricas SUS.

### **1.3 Organización de la memoria**

La memoria estará dividida en una serie de capítulos, en los cuales se irán detallando las diferentes fases que se han ido realizando.

**En el capítulo 2**, se detallará el estado del arte, en el que se contará en qué consiste la conducción eficiente. Además, se presentarán algunas aplicaciones similares y recursos útiles. Por último, se detallará por qué la elección de Android para nuestra aplicación.

**En el capítulo 3**, se explicará la metodología usada durante todo el proyecto, así como, las herramientas y aplicaciones usadas para llevarlo a cabo.

**En el capítulo 4**, se expondrán las fases de análisis y diseño de la aplicación, incluyendo los casos de uso que contendrá, así como el diagrama de clase, la maqueta y el diseño de la base de datos que utilizará la aplicación.

**En el capítulo 5**, se especificará la fase de desarrollo, donde explicaremos que se ha utilizado para realizar cada parte del proyecto, y soluciones que se han debido realizar a lo largo de la fase.

**En el capítulo 6**, se detallarán las pruebas que se han realizado sobre la aplicación desarrollada. Se realizarán pruebas unitarias, de validación, de verificación y de usabilidad, y se incluirá de igual modo una breve reflexión sobre los resultados de las mismas.

**En el capítulo 7**, se expondrán las conclusiones, tanto de la aplicación (a nivel de objetivos cumplidos), como a nivel personal.





## 2 Estado del arte

---

### 2.1 Conducción eficiente

Hoy en día, se hacen muchas campañas para que cada vez haya menos accidentes, por lo que es necesario tener una buena conducción. Para ello, la DGT[1] expone un documento en el que establece cuales son las medidas que se deben tomar para una conducción eficiente. Además, se especifican las ventajas que este modo de conducción aporta a los usuarios, es decir, por qué los conductores deberían optar por este paradigma en lugar de seguir con su conducción habitual. De igual manera, en dicho documento se exponen también una serie de consejos útiles para lograr una conducción más eficiente. De la lectura de dicho documento podemos extraer las siguientes ideas clave:

1. Al arrancar el motor no se debe pisar el acelerador.
2. En el inicio de la marcha se recomienda introducir cuanto antes la segunda velocidad.
3. En la medida de lo posible y siempre que la circulación del resto de vehículos lo permita, se aconseja utilizar marchas largas, de forma que el motor gire a menos revoluciones y se consiga de esta manera reducir el consumo de combustible.
4. Debe ser nuestro objetivo también, si el tráfico y la vía lo permiten, evitar las aceleraciones o deceleraciones bruscas. En caso de encontrar un obstáculo, es recomendable, si existe suficiente espacio entre nuestro vehículo y dicho obstáculo, levantar el pie del acelerador y dejar que la inercia disminuya, reduciendo paulatinamente la velocidad, o pisar el freno de forma suave. Del mismo modo, cuando nos dispongamos a aumentar nuestra velocidad y tengamos vía libre por delante, intentaremos no pisar fuertemente el acelerador, sino ir ganando velocidad de manera progresiva.
5. Mantener una distancia de seguridad adecuada con el vehículo que circula delante de nosotros, de forma que podamos reaccionar con mayor antelación en el caso de que se produzca una frenada repentina de dicho vehículo.
6. Cuando nos aproximemos a una curva, intentaremos anticiparnos al ángulo y características de la misma para adecuar la velocidad en consecuencia, evitando así cambios bruscos en la dirección o velocidad del vehículo.
7. Mantener la velocidad del vehículo dentro de los límites establecidos en el reglamento de circulación vigente.

8. Tratar de minimizar los cambios de carril a los mínimos indispensables para llegar de forma segura hasta nuestro destino, evitando los movimientos bruscos.
9. Intentar mantener una velocidad constante durante la mayor parte posible del trayecto.

Muchas de estas recomendaciones tienen que ver sobre todo con la velocidad, y sobre todo con el modo en que esta cambia a lo largo del tiempo (con la aceleración). Estos parámetros pueden verse reflejados fácilmente en una gráfica realizada por la aplicación, por lo que, utilizando la aplicación desarrollada, el usuario tendrá una manera de controlar en qué grado se ha acoplado a estas directrices de conducción eficiente, y tomar nota de los errores que debe enmendar para el futuro si desea mejorar la eficiencia de su forma de conducir.

## **2.2 Aplicaciones similares y recursos útiles.**

En la actualidad no existe mucha variedad de esta clase de aplicaciones. Tras una búsqueda intensiva solo se han encontrado algunos ejemplos de aplicaciones que comparten algunos objetivos con nuestra app *SafeCar*.

### **DriveSmart** [2], [3]



Esta aplicación analiza el comportamiento de los conductores en la carretera en base a una serie de parámetros, y emplea el refuerzo positivo mediante recompensas y logros para motivar a los conductores a mejorar su estilo de conducción y reducir su riesgo de tener accidentes de tráfico. Esta aplicación está recomendada para aquellos conductores noveles que necesitan confianza y seguridad en la carretera. A día de hoy, la aplicación continúa en fase de desarrollo.

## **OpenStreetMap[4]**



Se trata de una alternativa libre a aplicaciones como Google Maps. Esta aplicación es gratuita y pública, cualquier persona puede realizar aportaciones a este mapa. Tras investigar un poco en la aplicación, puede apreciarse que no dispone de todas las calles, y no podemos conocer con exactitud si los datos son correctos debido a que son los propios usuarios los que introducen toda la funcionalidad a la aplicación.

## **Waze[5]**



Es una aplicación social en tiempo real y de navegación asistida mediante el GPS. Fue comprada íntegramente por Google. Se comparte información vial sobre y el tráfico en tiempo real.

## **Velociraptor[6]**

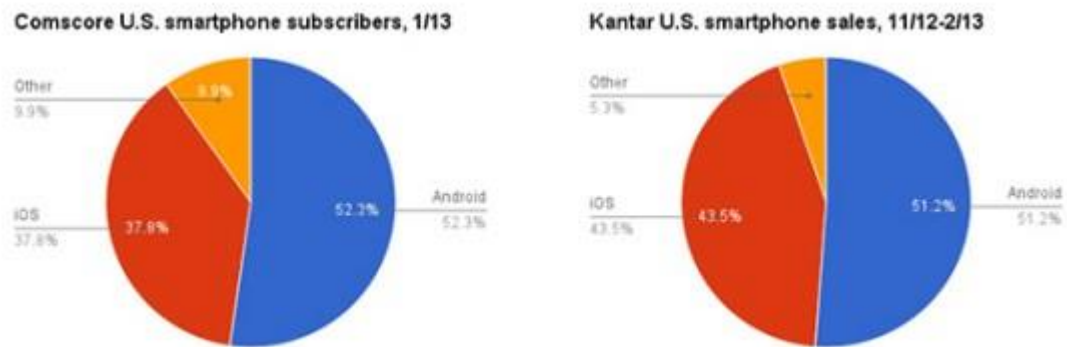


Es un widget flotante de diseño que se integra en aplicaciones como Google Maps y Waze aportando a estas la velocidad máxima permitida en la vía. En la sección de desarrollo se detallará por qué finalmente se decidió no usarlo.

## ***2.3 Elección del sistema operativo Android***

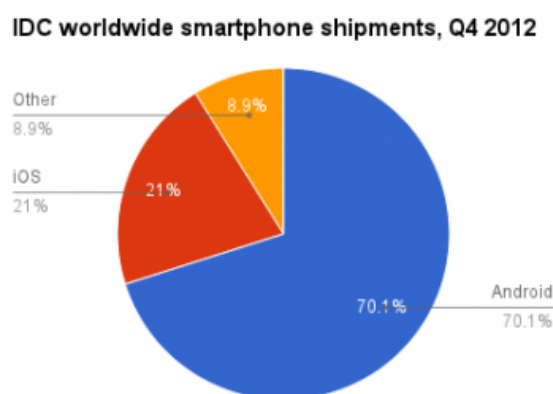
Hoy en día, la mayoría de la población tiene acceso a las últimas tecnologías. A la hora de comenzar el desarrollo de una nueva aplicación, una de las decisiones más importantes que debemos tomar es el entorno en el que se va a implementar. En primer lugar lo que debemos valorar es para qué tipo de dispositivo va a estar destinada esta aplicación. En este aspecto, se nos suele plantear la disyuntiva lógica de si plantear la aplicación para una Tablet o para un Smartphone, pues son los dos tipos de dispositivos móviles con acceso a internet más populares en la sociedad. La respuesta a esta cuestión es bastante rápida, debido a que normalmente la mayoría de la gente lleva su teléfono móvil durante el día con él. En cambio una Tablet, debido sobre todo a sus mayores dimensiones y peso, es menos probable que acompañe al usuario en la mayoría de sus tareas y trayectos cotidianos. Hay gente que la lleva por trabajo o por placer, pero es un elemento menos común en la rutina diaria que un Smartphone.

Una vez decidido el tipo de dispositivo que vamos a emplear, solo nos queda elegir el sistema operativo. En esta indecisión, tenemos dos candidatos; por un lado, Android y por el otro iOS, los cuales representan la práctica totalidad del mercado de Smartphones. Lo que se pretende es llegar al mayor número de gente posible. En [7] puede encontrarse la información resumida en la Figura 2-1.



**Figura 2-1: Gráficas Android o iOS realizadas en EEUU**

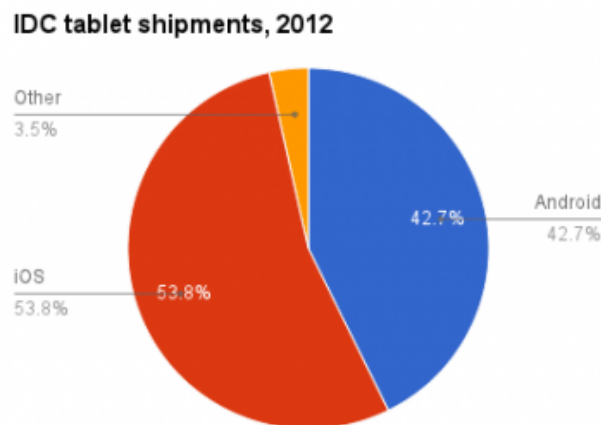
A la izquierda de la figura puede verse el estudio realizado por Comscore. En él puede verse cómo Android es usado un 52,3% frente a iOS que es usado un 37,8%. A la derecha puede verse el estudio realizado por Kantar. En éste, no hay gran diferencia, Android es usado un 51,2% y iOS un 43,5%. Estos estudios son de dos fuentes distintas para EEUU; si ahora miramos la gráfica de todo el mundo, nos daremos cuenta de que Android lleva mucha ventaja a iOS en número de ventas de móviles a nivel global (ver Figura 2-2).



**Figura 2-2: Gráfica Android o iOS mundial**

Teniendo en cuenta los datos sobre las estadísticas de ventas que hemos visto, todo parece indicar que si lo que queremos es llegar al mayor número de usuarios posibles, la mejor opción es realizar un desarrollo para Android, ya que es el sistema operativo móvil más utilizado a nivel global.

Como curiosidad, si en la primera decisión, es decir, al elegir teléfono móvil o Tablet, hubiésemos elegido Tablet, el resultado sería muy diferente. En Tablets, el sistema operativo más usado es iOS, como podemos ver en la Figura 2-3.



**Figura 2-3: Gráfica Android o iOS en tablets**

Pero tal y como se explicó anteriormente, un usuario es más fácil que disponga de un móvil que de una Tablet por lo que la decisión final será usar un teléfono móvil con sistema operativo Android.

## **2.4 Comentarios finales**

A lo largo de este capítulo se ha visto que aunque existen algunas aplicaciones pensadas para la conducción existen realmente pocas alternativas que se asemejen a lo que pretende la aplicación que se va a desarrollar.

Lo que hace diferente a *SafeCar* y además hace que destaque del resto de alternativas es que gracias a ella los conductores podrán disponer de una aplicación para el móvil que fomente hábitos de conducción eficiente y responsable haciendo así que se convierta en un reto estimulante para ellos.

A partir de todo lo que se ha detallado en esta sección, podemos concluir que existe una necesidad a cubrir que justifica inequívocamente el desarrollo de la aplicación.



## 3 Metodología y herramientas utilizadas

En esta sección se detallará la metodología seguida durante todo el proceso de desarrollo de la aplicación. Además, se explicarán las herramientas utilizadas para la implementación de *SafeCar*, así como los lenguajes de programación, el sistema operativo y la herramienta de control de versiones.

### 3.1 Metodología usada

En este apartado, se definirá la metodología usada para este proyecto, y una pequeña definición sobre cada una de las etapas de ésta. Esta parte es esencial, ya que determinará el conjunto de métodos a utilizar para que la actividad esté optimizada y formalizada.

Para este TFG se ha utilizado el Proceso Unificado de Desarrollo o Proceso Unificado de Rational (RUP)[8], que consiste en no establecer un sistema con pasos fijados, sino en adaptar la metodología al contexto y a las necesidades del proyecto que se está realizando. Para decirlo de otra manera, no es necesario que una fase del proyecto vaya justo después que la otra, si no que antes de terminar una ya se ha empezado la siguiente, por lo que puede haber solapamientos. En la Figura 3-1 (extraída de [9]) puede observarse el solapamiento del que hablamos.

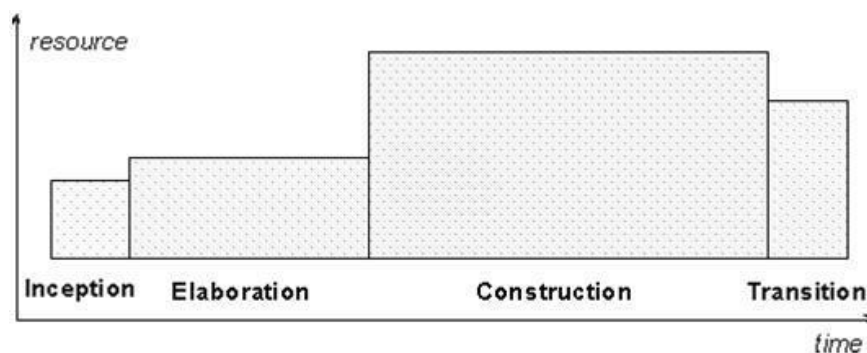


**Figura 3-1: Ejemplo de planificación**

Este modelo tendrá las mismas fases que cualquier otra metodología, toma de requisitos, análisis, diseño, implementación y pruebas, las cuales se muestran en la foto anterior. Podemos tener más fases pero estas son las que se han realizado en este proyecto. Esta metodología divide el proceso en cuatro etapas en las cuales se irán realizando iteraciones dependiendo de las necesidades del proyecto, es decir, dependiendo del trabajo que requiera cada etapa. Las cuatro etapas son las siguientes:

1. **Fase de iniciación:** en esta fase se recogerán los requisitos del usuario, además de evaluar los posibles riesgos del proyecto y poder fijarse unos hitos.
2. **Fase de elaboración:** en esta fase se seleccionarán los casos de uso que definirán el proyecto, haciendo así un diseño inicial del proyecto.
3. **Fase de construcción:** en este paso se implementará la funcionalidad del sistema y, para ello, se deberá utilizar lo realizado en las etapas anteriores o lo que se esté llevando a cabo al mismo tiempo de la etapa anterior.
4. **Fase de transición:** esta fase tiene que asegurar que los usuarios tienen el sistema o proyecto final disponible y se establecerá un plan de mantenimiento durante el tiempo acordado en la fase inicial con el cliente.

Cada una de estas fases tiene una duración determinada, no obstante, como no todas las fases requieren la misma cantidad de esfuerzo para llevarlas a cabo, es habitual que una determinada fase pueda ser más larga que la anterior. En la Figura 3-2 (extraída de [10]) podemos ver que la fase de construcción suele ser más costosa que el resto.



**Figura 3-2: Diagrama duración fases de la metodología empleada**

Se ha elegido este modelo debido a que permite realizar cambios a la vez en toda la arquitectura según se vaya necesitando en procesos posteriores. De esta manera, todo queda mejor definido para el usuario final.



### **3.2 Plataformas y tecnologías**

Para el desarrollo de la App se han hecho uso de una serie de tecnologías y plataformas que se pasan a listar a continuación:

#### **Android Studio**



Android Studio[11]–[13] es un entorno de desarrollo integrado para la realización de aplicaciones Android. Se ha elegido esta plataforma debido a que es la más común y mejor preparada para llevar a cabo proyectos Android. Además, se ha utilizado durante la carrera y se ha propuesto como una aplicación recomendable para estos temas.

#### **Bitbucket**



Bitbucket [14] es un sistema de control de versiones distribuido accesible desde la web, para proyectos que utilizan sistemas de control de versiones como Mercurial y Git.

En este caso, se ha utilizado Git. Gracias a esta plataforma ha sido mucho más fácil la organización y ver con mayor precisión los cambios en el código entre diferentes versiones del software desarrollado, de forma que se facilita enormemente la tarea de detección de errores en el código para su posterior corrección.

#### **SQLite**



Es una plataforma para la gestión de bases de datos en lenguajes como SQLite. Se ha usado para poder controlar los cambios que se realizaban en la base de datos y poder comprobar si eran correctos. Además se ha utilizado a la hora de crear el modelo de datos para nuestra base de datos, pudiendo comprobar gracias a esta herramienta si se realizaba bien, es decir, si tenía todos los campos esperados.

#### **Java**



Java es un lenguaje de programación de propósito general y orientado a objetos, que fue diseñado para tener tantas dependencias de implementación como fuese posible. La mayoría de aplicaciones Android son programadas en Java, por lo que parecía lógico utilizar este lenguaje para nuestro desarrollo.

## XML



XML (eXtensible Markup Language) [15] es un lenguaje de marcado que define un conjunto de reglas para codificar documentos en un formato que sea al mismo tiempo legible para los seres humanos y para las máquinas, desarrollado por el W3C (World Wide Web Consortium). Este será utilizado para poder realizar la interfaz del usuario en Android.

## SQLITE



SQLite [16], [17] es un sistema de gestión de bases de datos relacionales. Se ha elegido debido a que es el más común en los desarrollos para Android, además de ser uno de los más rápidos al ejecutar consultas sobre la base de datos, lo cual se ajusta a las necesidades de nuestro proyecto.

### **3.3 Sistema operativo: Android**



Android [18] es un sistema operativo móvil desarrollado por Google, basado en su mayor parte en una versión modificada del kernel de Linux, diseñado principalmente para dispositivos móviles con pantalla táctil, como teléfonos inteligentes y tabletas. En los últimos años, además, se han diseñado versiones de este Sistema Operativo para otros tipos de dispositivos inteligentes como relojes o televisores.

Como Android es un Sistema Operativo que presenta una gran fragmentación entre las diferentes versiones que utilizan la multitud de dispositivos que lo llevan instalado a lo largo y ancho del globo, surge la necesidad de especificar, a la hora de desarrollar una aplicación para este Sistema Operativo, cuáles serán las versiones del SDK de Android mínima y objetivo, respectivamente.

En nuestro proyecto, la versión mínima del SDK que soportará la aplicación final será la versión 15, correspondiente a Android 4.0.3 Ice Cream Sandwich, mientras que la versión objetivo para la que se va a desarrollar será la versión 26 del SDK, correspondiente a la versión de Android 8.0.0 Oreo, que es una de las más actuales, y la que se supone que acogerá un mayor número de usuarios de aquí a un tiempo.

## 4 Análisis y Diseño

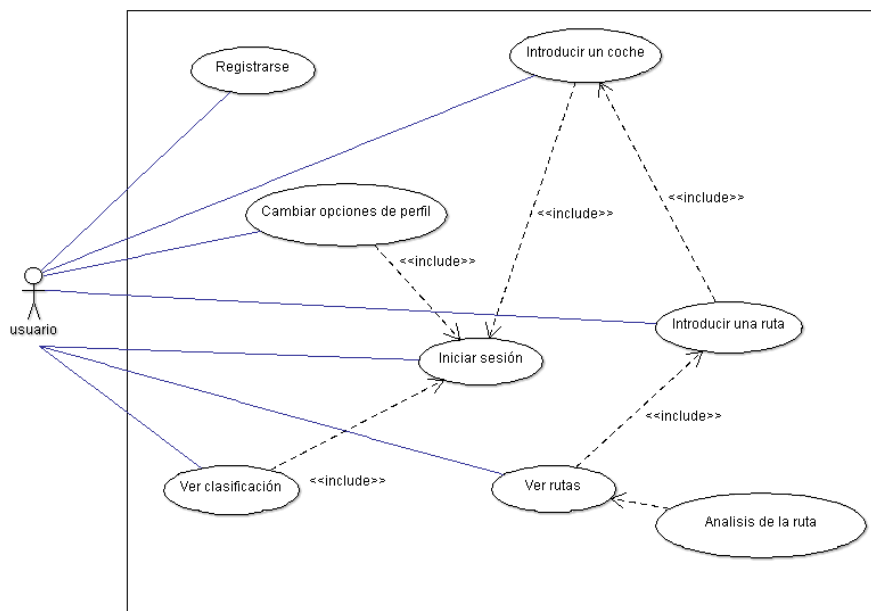
### 4.1 Introducción

En este capítulo, mostraremos con detalle la fase de análisis, siendo esta una de las más importantes para el proceso de desarrollo software. Para ello, mostraremos el diagrama de casos de uso [19] y una explicación de cada uno de ellos de manera individual. Además, se detallarán los requisitos que tendrá la aplicación.

Expondremos también la fase de diseño, en la cual se engloba el diagrama de clases de la aplicación [20], de forma que se establezcan las bases a seguir en la etapa posterior de desarrollo. Por otro lado, se presentará el diseño de la base de datos que utilizaremos para guardar toda la información necesaria y, finalmente, la maqueta, que nos dará una idea gráfica de los requisitos que debe alcanzar nuestra aplicación.

### 4.2 Casos de uso

En este apartado, mostraremos un diagrama con todos los casos de uso para ver qué relación tendrán unos con otros y, tras ello, mostraremos un breve resumen de cada uno de ellos. Como podemos observar en el diagrama de la Figura 4-1, la aplicación solo tendrá un rol, el cual será el del usuario que descargue la aplicación.



**Figura 4-1: Diagrama de casos de uso**

### **4.2.1 Caso de uso de registro**

Para poder empezar a utilizar la aplicación, el usuario deberá registrarse de manera obligatoria. Por esta razón, después de la instalación de la aplicación, será necesario que el usuario se registre introduciendo su correo electrónico y contraseña. Una vez que el usuario se haya registrado, no será necesario que vuelva a realizar este paso. Se deberá introducir por duplicado las contraseñas, para comprobar que la contraseña que el usuario ha introducido es realmente la que desea introducir. En el caso de que ya haya un usuario con el mismo nick o e-mail, o las contraseñas no sean iguales, se emitirá un mensaje de error.

### **4.2.2 Caso de uso de inicio de sesión**

Este caso de uso se realizará cuando el usuario haya terminado el anterior. Para iniciar sesión, solo se deberá introducir el correo electrónico y contraseñas anteriormente definidos en el registro. Si se introduce un correo electrónico o una contraseña incorrectos, se mostrará un mensaje de error para que el usuario sea consciente de que algo ha ido mal.

### **4.2.3 Caso de uso de cambiar opciones de perfil**

El usuario tendrá a su disposición un perfil, en el cual podrá cambiar ciertas opciones, tales como su nombre, apellidos, contraseña y el país en el que se encuentra. Se ha decidido que el usuario no pueda cambiar ni el correo electrónico ni el nick, debido a que estos serán identificadores únicos usados en la aplicación, ya sea tanto para la autenticación como para la clasificación.

### **4.2.4 Caso de uso de introducción de un coche**

Un usuario puede disponer de varios coches, por lo que no necesariamente todos los trayectos los realizará con el mismo, es decir, podrá utilizar un coche diferente para hacer un mismo trayecto. Por esta razón el usuario tendrá la opción de introducir varios coches, teniendo que seleccionar de cada uno de ellos, el año de fabricación, la marca, el modelo y el tipo de transmisión del vehículo (es decir, si es automático o manual). Además, deberá ponerle un nombre identificativo, de forma que el usuario pueda después reconocer qué coche está utilizando. Con esto, el usuario sabrá con qué coche ha realizado cada trayecto. Por otro lado, será imprescindible introducir un coche para poder empezar a realizar trayectos.

#### 4.2.5 Caso de uso de introducción de trayecto

Cuando se haya introducido al menos un coche, el usuario podrá introducir una ruta nueva o podrá reutilizar otra anteriormente realizada, es decir, podrá utilizar el mismo nombre para varios trayectos. Cuando el usuario decidida comenzar el trayecto, deberá seleccionar antes el coche con el que va a realizarlo. Una vez introducida la ruta, nos pondremos en marcha.

Durante el trayecto, aparecerá en pantalla la calle en la que se encuentra el conductor, o la vía, la velocidad que lleva en todo momento el vehículo, así como el tiempo transcurrido desde el inicio de la ruta. Una vez hayamos finalizado el viaje, el usuario deberá indicarlo mediante la pulsación de un botón. Para poder realizar este caso de uso, es indispensable tener internet para poder acceder a la calle en la que nos encontramos y un sistema GPS para poder acceder a la localización.

#### 4.2.6 Caso de uso de ver los trayectos

El usuario tendrá la posibilidad de ver los trayectos que ha realizado. En la pantalla principal se mostrarán los cinco últimos trayectos realizados, de tal manera que el usuario podrá acceder a los detalles de los mismos rápidamente. Por otro lado, el usuario podrá ver las rutas que ha realizado con cada coche, y al igual que en la opción anterior podrá acceder al análisis del trayecto.

#### 4.2.7 Caso de uso de análisis del trayecto.

Para analizar la ruta, lo que haremos será mirar ciertos parámetros:

- **La velocidad:** se cogerá la velocidad máxima de la carretera y la velocidad a la cuál va el conductor. Si la velocidad a la que se ha circulado durante el trayecto es superior a la velocidad marcada como máxima, determinaremos que el usuario ha cometido una infracción.
- **Frenazos y aceleraciones:** se mostrará gráficamente la medida en la que el conductor realiza aceleraciones y deceleraciones bruscas, de forma que el usuario pueda actuar en consecuencia y tratar de corregir este comportamiento al volante poco eficiente para el próximo trayecto.
- **Velocidad constante:** llevando una velocidad constante la contaminación disminuye por lo que es beneficioso para el medio ambiente y para la conducción.

Estos parámetros se mostrarán en una gráfica, en la cual se podrá observar cómo ha ido el viaje. Se podrán mostrar varios trayectos a la vez, siempre y cuando estos hayan sido etiquetados como el mismo (es decir, ambos trayectos tengan el mismo nombre). Además, se mostrarán las multas más relevantes que podría tener un conductor si un radar detectase su paso a esa velocidad por esa vía, en caso de que la velocidad del vehículo fuese superior a la máxima establecida según el reglamento para ese tipo de vía.

#### 4.2.8 Caso de uso de la visualización de la clasificación.

Tendremos una sección en la cual podremos ver una clasificación, es decir, saber quién es el mejor conductor a nivel de aplicación. Para establecer quién es el mejor conductor a nivel de aplicación lo que haremos será ver los puntos que tiene cada usuario, es decir, el que mayor número tenga, ese será el primero y así sucesivamente. Durante un trayecto, se guardará el número de veces que el usuario comete una incidencia, es decir, cuando la velocidad a la que circula es mayor que la velocidad máxima permitida en la vía. Esto puede llevar a que el usuario tenga demasiadas infracciones, por lo que en lugar de restar puntos por cada incidencia que se registre, las iremos agrupando de forma que, si el usuario no ha realizado ninguna infracción, se le asignarán 100 puntos por trayecto. A partir de ese valor, su puntuación irá disminuyendo a medida que vaya subiendo el número de infracciones realizadas, tal y como se muestra en la siguiente tabla:

Número de infracciones	Puntos obtenidos
10	80
20	60
30	50
40	40
50	30
60	20
70	10
100	0

**Tabla 1 : Puntos por infracciones cometidas**

### 4.3 Diagrama de clases

En este apartado mostraremos el diagrama de clases simplificado de la parte de la lógica de la aplicación, en Lenguaje Unificado de Modelado (UML), en el cual se describe la estructura estática que tendrá el sistema, mostrando para ello sus clases. Este diseño es el inicial, pero en la implementación se ha ido aumentando dependiendo de la necesidad del momento.

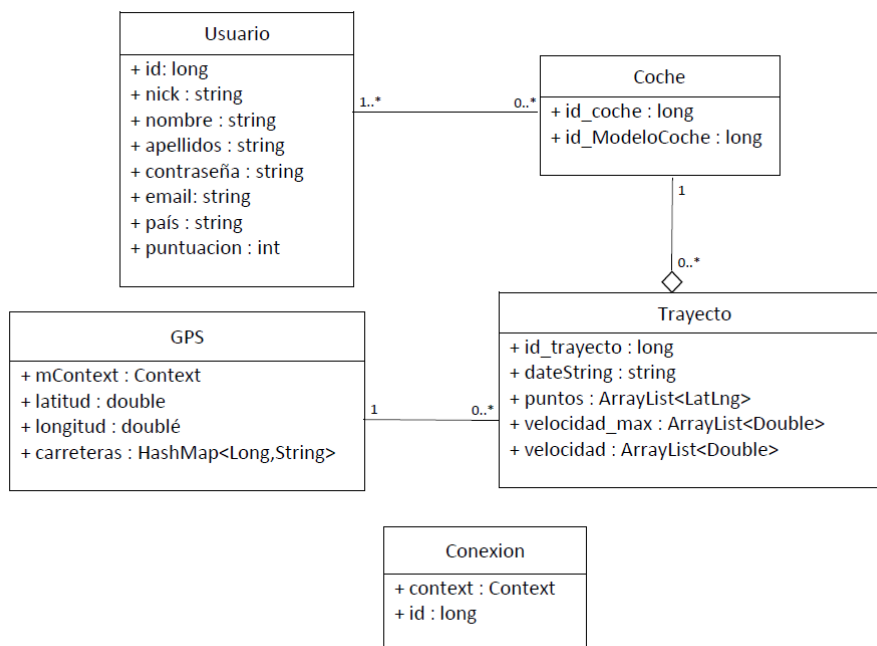


Figura 4-2: Diagrama de clases simplificado

### 4.4 Maquetas de la interfaz de usuario

Cada una de las pantallas de la maqueta, podemos decir que será un caso de uso de los nombrados anteriormente en la fase de análisis (Apartado 4.2). Cada caso de uso, en Java, tendrá su parte lógica y su parte gráfica (la referente a la interfaz de usuario).

La maqueta se detalla en el Anexo A, especificando la navegación entre pantallas para facilitar el entendimiento de la aplicación.

Con la maqueta conseguimos una visualización previa de la aplicación, de forma que tengamos un primer concepto de lo que será. Esto nos permitirá ahorrar tiempo en la fase de desarrollo donde no tendremos que pensar cómo llevar a cabo la implementación de cada uno de los requisitos, puesto que la maqueta ya nos proporciona algunas ideas de cómo realizarlo.

## 4.5 Diseño de la base de datos

En esta sección se presentará el diseño de la base de datos que se utilizará en la aplicación. Para ello, se ha utilizado un modelo entidad relación[21], para el modelado de datos que permitirá representar las entidades relevantes del sistema de información, junto con sus relaciones y propiedades. En la Figura 4-3, podemos ver cómo ha quedado el diagrama entidad-relación.

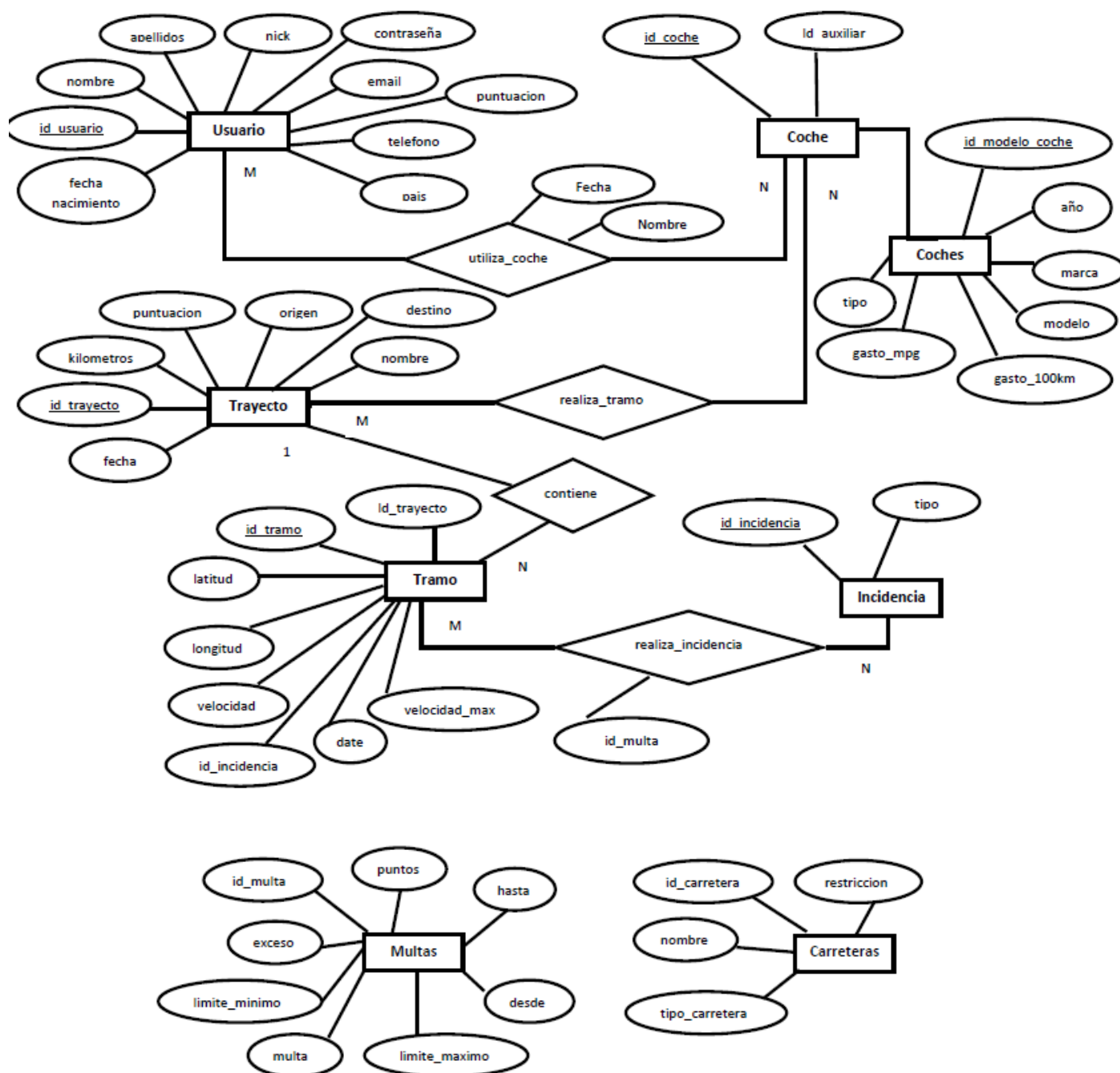


Figura 4-3: Modelo entidad-relación



Para la realización de este diseño, lo primero que se pensó fue cuáles eran los elementos más relevantes. Tras una breve reflexión, se determinó que dichos elementos serían: los usuarios, los coches y los trayectos.

El usuario necesita identificarse, y el sistema necesita identificarlo, por lo que cada usuario tendrá un identificador, de forma que la aplicación pueda acceder a la base de datos en busca de la información necesaria sobre este. Además, el usuario necesita saber cómo identificarse: para ello, en el registro se introduce el nick, el correo y la contraseña. El resto de datos serán menos relevantes, pero de igual modo, necesarios para poder tener el perfil completo.

Como la aplicación desarrollada se basa en los datos recogidos sobre las coordenadas GPS de las ubicaciones por las que va pasando el dispositivo en el que está instalada, dependeremos de un vehículo que transporte a dicho dispositivo para que pueda ir recogiendo dichos datos, probando así todas y cada una de las funcionalidades presentes en la aplicación. Ante esta situación, cobra una gran importancia la manera en la que se relacionan los coches con los usuarios dentro de la aplicación.

Para llevar a cabo esta tarea, dispondremos de la relación intermedia *utiliza\_coche*, que representará dicha relación entre un vehículo y el usuario que lo utiliza (lo conduce). De igual modo, existe la posibilidad de que varios usuarios asignen el mismo nombre a uno de los vehículos que utilizan, de modo que se hace necesaria una tabla intermedia en la que se haga patente la relación entre ese coche y ese usuario gracias a sus identificadores y, además, a la fecha en la que se introduce el coche y el nombre del mismo en el sistema.

Por otro lado, necesitamos tener todos los coches en la base de datos para que el usuario pueda elegir uno. Para cumplir con este cometido, se ha creado la tabla externa *modelos\_coche*, en la que tendremos algunos coches junto con su consumo. Estos coches han sido extraídos de La Agencia de Protección Medioambiental de Estados Unidos (EPA [22]). Se han incluido todos los coches de los años 2016, 2017 y 2018. Por otro lado, se ha modificado la base de datos (extrayendo los camiones de esta) y se han cogido los elementos que se han considerado necesarios, tales como: modelo, año, marca y consumo. Se debatió también la idea de utilizar una base de datos española, con el objetivo de que la mayoría de los coches fuesen reconocidos por los usuarios. Sin embargo, no se encontró ninguna base de datos que cumpliera con los requisitos que se buscaban. Por esta razón finalmente se decidió utilizar la de la EPA, que incluía también algunos coches bastante comunes entre la población de España.

Por otra parte, el elemento fundamental para este trabajo fin de grado son los trayectos que el usuario va a realizar. Un trayecto va a estar realizado por un coche y usuario. Por esa razón, el trayecto tendrá relación con el coche y este con el usuario. En su momento, se pensó poner una relación entre usuario y trayecto. Esta idea fue descartada debido a que la redundancia podría traer problemas a la hora de tener relaciones duplicadas. Además, un mismo coche puede realizar las mismas rutas varias veces, por lo que esta relación se convertirá en una nueva tabla.

El trayecto estará constituido por un origen, un destino, un nombre, la fecha en la que se realizó y los kilómetros que recorrió. El nombre del trayecto servirá al usuario para diferenciar este del resto. No obstante, como se ha comentado anteriormente, varios trayectos podrán tener el mismo nombre.

Pero el planteamiento no acaba aquí, el trayecto está compuesto por muchos puntos por los que irá pasando el conductor. A cada uno de esos puntos lo llamaremos tramo. Cada punto representa una coordenada GPS, por lo que necesitaremos la latitud, la longitud, la velocidad del vehículo, la velocidad máxima de la vía, la fecha en la que se registra ese punto y saber si ha tenido lugar una incidencia en dicho punto o no. Como sabemos, un punto GPS será único para cada trayecto, por lo que cada tramo tendrá el identificador del trayecto al que corresponde.

Por último, tendremos que tener en cuenta las incidencias. Para este proyecto, sólo tendremos como incidencia superar la velocidad máxima de la carretera. Como muchos tramos pueden tener la misma incidencia, de nuevo se necesita una tabla auxiliar, la cual contendrá la información necesaria para relacionar el tramo de la vía en el que se comente una incidencia con la propia incidencia, así como con la multa que acarrearía, para cada punto en el que se ha superado la velocidad máxima de la vía por la que se circula.

Aparte de las tablas anteriormente nombradas, tendremos dos tablas independientes pero necesarias para nuestra base de datos. Por un lado, la tabla “Multas”, que contendrá la sanción y los puntos que el usuario deberá asumir en el caso de cometer una infracción. Además, se incluye el intervalo de velocidades para el que se contempla dicha multa, y en qué medida deberá ascender esa velocidad para que constituya una u otra sanción. Para realizar esta tabla, se ha accedido a la DGT para ver las sanciones que aplican en cada caso, y lo que se ha encontrado se muestra en la tabla 2 (extraída de [23]). Cabe destacar que esta tabla no es la que se ha utilizado, si no que se trata de una versión modificada para facilitar su lectura y comprensión.

Límites		30	40	50	60	70	80	90	110	120	Multa	Puntos
Excesos de velocidad	Grave	31	41	51	61	71	81	91	111	121	100	0
		50	60	70	90	100	110	120	140	150		
		51	61	71	91	101	111	121	131	141	300	2
		60	70	80	110	120	130	140	150	160		
		61	71	81	111	121	131	141	151	161	400	4
		70	80	90	120	130	140	150	160	170		
	71	81	91	121	131	141	151	161	171	500	6	
	80	90	100	130	140	150	160	170	180			
Muy Grave	81	91	101	131	141	151	161	171	181	600	6	

**Tabla 2 : Puntos y sanción por velocidades**

Por otro lado, se ha definido la tabla de carreteras, la cual contiene el nombre de las autovías, carreteras autonómicas y nacionales de toda España, indicando en la base de datos, tal y como se ha dicho anteriormente, el nombre, el tipo de vía y si se conoce alguna restricción de la carretera. En el apartado de desarrollo se explicará con detalle el porqué de la utilización de esta tabla en la base de datos.



## 5 Desarrollo

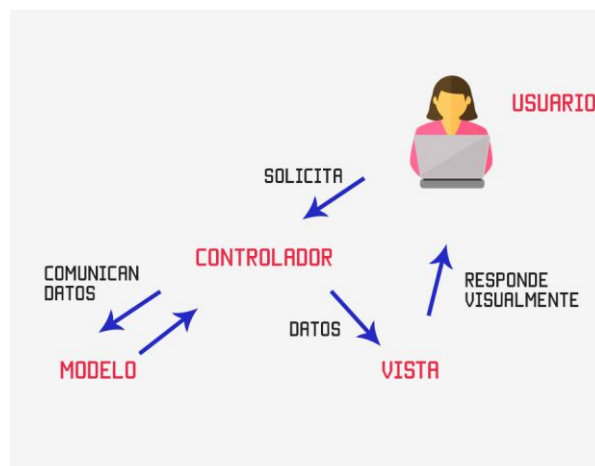
---

### 5.1 Introducción

En este capítulo detallaremos la fase de desarrollo de la aplicación, es decir, cómo se ha ido implementando y qué problemas se han ido encontrando y, por consiguiente, qué alternativas se han elegido para solucionarlos. Esta fase ha sido la más larga de todo el proyecto, ya que se han necesitado recursos externos, los cuales era la primera vez que se utilizaban.

### 5.2 Patrón de diseño utilizado

Para el desarrollo de *Safecar*, cabe destacar que se ha usado uno de los patrones más importantes de la arquitectura del software, este es el patrón Modelo-Vista-Controlador (M-V-C) [24]. Este patrón pretende separar la lógica de la aplicación de la lógica de la vista, es decir, separarla de la interfaz de usuario. Para entenderlo bien, podemos ver la Figura 5-1.



**Figura 5-1: Resumen MVC**

El usuario solicita una acción, la cual se le pasará al controlador. Éste pedirá los datos necesarios al modelo. Una vez que el controlador tenga los datos, se le pasarán a la vista, de tal manera que el usuario podrá ver el cambio.

### **5.3 Cuestiones y detalles de implementación de la base de datos**

La base de datos es un elemento imprescindible en *SafeCar*, debido a que sin ella no se podrían guardar todos los datos necesarios para el análisis de la conducción eficiente. Al comenzar a desarrollar esta parte, se utilizó la librería JDBC [25]. El proceso se realizó de la siguiente manera: se establecía el nombre de la base de datos, y a partir de ahí se creaba, se añadían o se eliminaban los datos necesarios. Se disponía de un fichero sql en el cual se encontraban todos los comandos necesarios para crear las tablas, de forma que al ejecutarlo se crearán las mismas.

Tras probarlo, en un teléfono móvil no funcionaba mientras que en local, es decir, desde Android Studio, sí. Esto es debido a que Android tiene una clase especial para realizar las consultas, a la cuál se le deberá indicar lo que se quiere buscar mediante sus parámetros, y esta devolverá la consulta realizada. Debido a esto, hubo que cambiar todo el código para utilizar la clase *SQLiteOpenHelper* [26].

La ventaja que tiene este sistema es que, una vez descargada la aplicación en el teléfono móvil, la primera vez que se desee realizar una consulta, este llamará al método de crear la base de datos y creará todas las tablas que se hayan especificado. Sin embargo, el resto de veces, tendrá la base de datos almacenada y simplemente tendrá que acceder a ella cogiendo su referencia. La base de datos se guarda en la memoria interna del teléfono, pero se hace de forma que no esté visible para el usuario final de la aplicación. No obstante, para poder comprobar que el funcionamiento de dicha base de datos es el correcto, desde Android Studio sí que se podrá acceder a dicho directorio, facilitando las tareas de *debugging*, pudiendo incluso descargarla a otra ubicación.

Al crear la base de datos, lo primero que deberemos hacer será crear las tablas necesarias para el funcionamiento de la aplicación según se describen en el modelo de datos. Para ello, en la clase en la que se implementa la conexión con la base de datos, se definirán dichas tablas como constantes, y se ejecutarán las consultas una a una desde ella.

Hay que tener en cuenta, que tras crear la estructura, hay tablas que necesitan que se les introduzca información (tablas como *MULTA*, *CARRETERAS* y *COCHES*). Por lo que, en Android, se debe crear una carpeta llamada “assets”, que será donde se guarden todos estos ficheros para poder volcarlos en la base de datos. Leeremos cada una de las líneas del fichero y las iremos introduciendo en la base de datos. Algunas tablas tienen una gran dimensión, por esa razón, la primera vez que se realice cualquier consulta a la base de datos la aplicación tardará en responder, debido a que se estarán introduciendo todos los valores anteriormente citados.

## **5.4 Cuestiones y detalles de implementación del sistema GPS**

Otro de los grandes puntos de este Trabajo Final de Grado es el uso del GPS del teléfono móvil para poder saber en cada momento a qué velocidad conduce el usuario en cada momento y poderlo comparar así con la velocidad máxima de la vía. Este ha sido uno de los puntos más complejos de todo el sistema. Además, sin éste la aplicación no podría cumplir ninguno de los objetivos propuestos. Como se ha nombrado varias veces, el móvil deberá tener GPS y conexión activa a internet, por lo que si el usuario no tiene el GPS conectado, no podrá realizar ningún trayecto. De este modo, para realizar un trayecto comenzaremos comprobando si está activo. En caso afirmativo, podremos pasar a introducir un trayecto.

Se ha realizado una clase para el tratamiento de los datos GPS. Para ello, nos ayudaremos de la clase `Service` y la interfaz `LocationListener` [27]. Esta última nos ayudará a recoger las coordenadas GPS cada vez que el usuario cambie de posición.

Antes de realizar lo anterior, se debe establecer cada cuanto tiempo (o distancia) queremos actualizar el GPS: en nuestro caso se ha establecido que se actualice cada segundo o cada vez que el usuario haya recorrido 15 metros. Se han fijado estos parámetros debido a que si se realiza un trayecto por ciudad, será necesario recoger más coordenadas, ya que el usuario estará accediendo a más calles.

Una vez establecido el tiempo y la distancia, debemos establecer si la ubicación se va a medir a través del GPS o de la conexión a internet. Si aún no existe ninguna posición almacenada, es decir, es la primera, la guardaremos como ubicación inicial. Para las posteriores veces, al implementar `LocationListener`, cada vez que el usuario cambie de posición este llamará a una función y en ella será donde haremos todo el tratamiento del trayecto. En este calcularemos varias cosas:

- Si la velocidad que lleva el usuario en ese momento es mayor que la de la carretera, tendremos una incidencia. Aquí, nos encontramos con que no es posible obtener la velocidad máxima de todas las carreteras de manera gratuita y sin necesidad de otras aplicaciones. Existe *Velociraptor* la cual nos ofrece dicha información; no obstante, se ha considerado que la utilización de esta herramienta sobrepasaba los límites del alcance de este proyecto por su difícil acceso.

Después, se tuvo en cuenta la utilización de Google Maps [28], aparte de que no está disponible para España, se debía tener una cuenta Premium, la cual conllevaba una cuota mensual. Por ello, la decisión final, fue realizar en la base de datos una tabla, en la cual se introdujesen todas las autovías, carreteras nacionales y comarcales de España [29]–[31]. En ella además, se establecería el tipo de carretera que es y en el caso de que se conozca la velocidad real en ciertas carreteras se pone una restricción para utilizar esa velocidad. Se asume que en todas las autovías se podrá circular a 120 km/h, en las carreteras nacionales a 100 km/h y en las carreteras comarcales a 90 km/h. Dentro de ciudad o pueblos se podrá circular como máximo a 30 km/h y en las carreteras comarcales a un máximo de 90 km/h. En el caso de que el usuario exceda la velocidad se introducirá una incidencia y además, en caso de que haya una multa también se guardará.

Por otro lado, es necesario pedir que el usuario dé permisos a la aplicación para acceder a la ubicación. Si no lo hace, la ubicación que devolverá será la del router o repetidor más cercano y no la real del usuario. Este ha sido uno de los contratiempos más difíciles de solucionar debido al desconocimiento de la existencia de dichos permisos.

- Para saber en la vía que se encuentra el usuario, a partir de la latitud y la longitud dadas por la posición, podremos establecer ésta mediante la clase Geocoder [32].
- Además, se calculará el tiempo del trayecto del usuario, desde el momento en el que se recoge la primera coordenada GPS hasta el momento en el que se encuentre el usuario.


## **5.5 Interfaz de usuario final**

En este apartado se mostrará cual ha sido el diseño final de cada una de las pantallas de la aplicación y se comparará con las maquetas realizadas para explicar por qué no se realizó de esa manera. Además, se realizará una visita por todas las pantallas. La maqueta se encuentra, tal y como se ha mencionado anteriormente, en el apartado 4.4., en el Anexo A, por lo que en esta sección, se hará referencia a las figuras de dicho anexo.



5.5.1 Guía de usuario

SafeCar



SafeCar

Usuario

Contraseña

He olvidado mi contraseña

ENTRAR

No tienes cuenta. ¡Regístrate!

1


SafeCar

¿Has olvidado la contraseña?

Introduce tu e-mail y te enviaremos una nueva contraseña

e-mail

ENVIAR

 SafeCar

2

SafeCar

Registro


Nick

Correo Electrónico

Contraseña

Repite Contraseña

ENVIAR

 SafeCar

3

SafeCar

Registro

Virginia


virginia2@uam.es

\*\*\*

\*\*\*

ENVIAR

Las contraseñas no son iguales

 SafeCar

4

SafeCar

Registro

virginia


virginia@uam.es

\*\*\*

\*\*\*

ENVIAR

Ya existe un usuario con ese nick

 SafeCar

5

SafeCar


¿Has olvidado la contraseña?

Introduce tu e-mail y te enviaremos una nueva contraseña

virginia@uam.es


ENVIAR

No existe un usuario con ese correo

 SafeCar

6

SafeCar



SafeCar

virginia@uam.es

Contraseña

He olvidado mi contraseña

ENTRAR

La contraseña o el usuario no son correctos

No tienes cuenta. ¡Regístrate!

7

SafeCar

Aún no ha realizado ningún trayecto

8

SafeCar

Perfil

Introducir coche

Introducir trayecto

Clasificación

Ver trayectos

Aún no ha realizado ningún trayecto

9

SafeCar

### Perfil del usuario

Nombre

Apellidos

País

Contraseña

Repite Contraseña

EDITAR

 **10**

SafeCar

### Perfil del usuario

Nombre

Apellidos

País

Contraseña

Repite Contraseña

GUARDAR

 **11**

SafeCar

### Perfil del usuario

Nombre

Apellidos

País

Contraseña

Repite Contraseña

GUARDAR

Las contraseñas son distintas

 **12**

SafeCar

### Introducir coche

Nombre del coche

Año

Marcas

Modelos

Tipo

GUARDAR

 **13**

SafeCar

### Introducir coche

AudiA4

Año

Marcas

Modelos

Tipo

GUARDAR

Debe seleccionar un coche

 **14**

SafeCar

### Introducir coche

AudiA4

2017

Audi

A4

Manual

GUARDAR

Ya hay un coche registrado con el mismo nombre

 **15**

SafeCar

Aún no ha realizado ningún trayecto

Para realizar un trayecto debe encender el GPS

**16**

SafeCar

Aún no ha realizado ningún trayecto

Debe introducir un coche

**17**


SafeCar

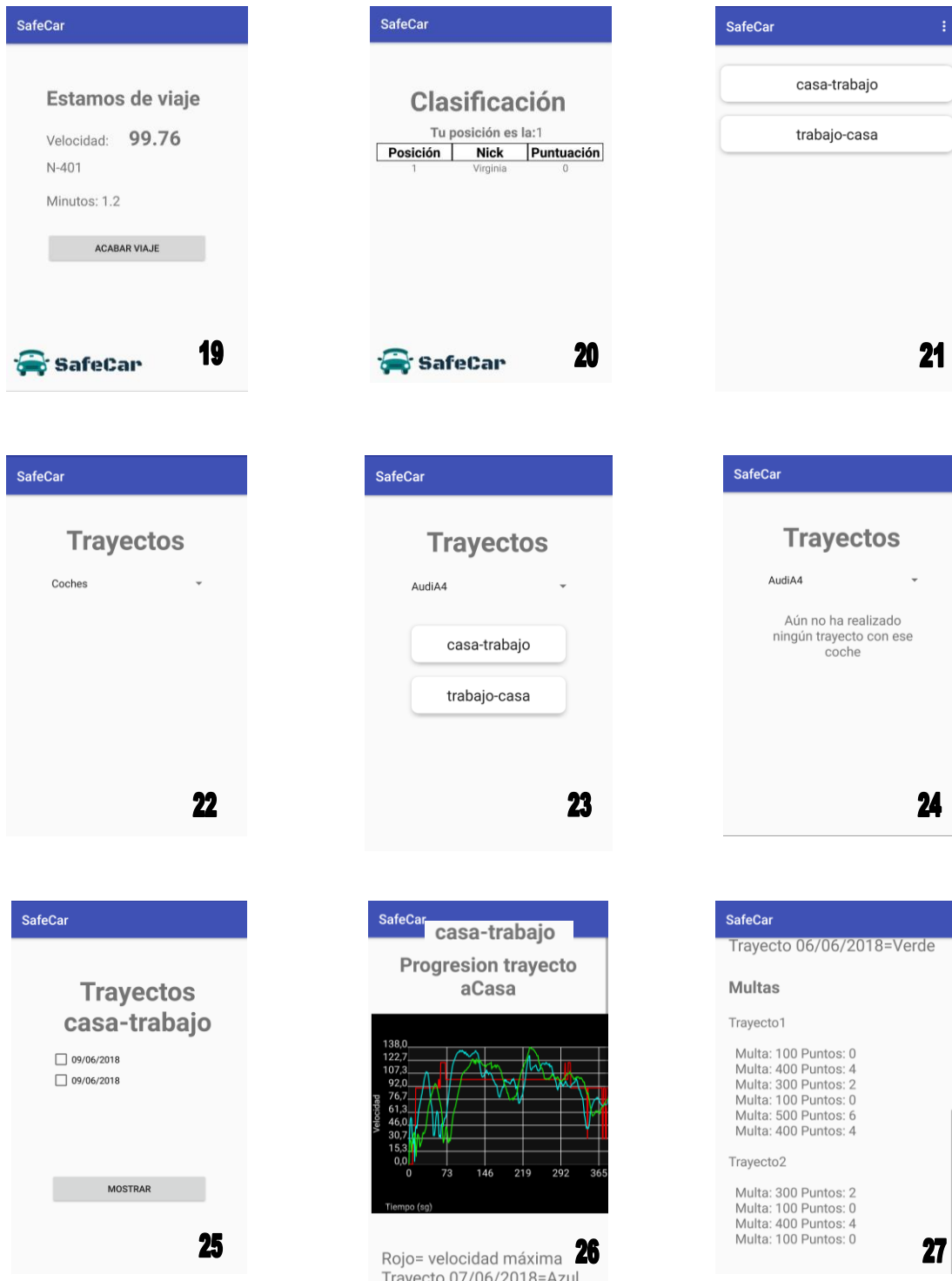
### Introducir una ruta

Nombre del trayecto

AudiA4

COMENZAR

 **18**



**Figura 5-2: Capturas de la interfaz del usuario**

Cuando los usuarios descarguen la aplicación, la primera pantalla que encontrarán será la etiquetada con el número 1 de las anteriores. Aquí el usuario podrá realizar varias acciones. En primer lugar, deberá estar registrado, para ello se pulsará sobre “No tienes cuenta. ¡Regístrate!”, con lo que pasaremos a la pantalla número 3. En el caso de que el nick del usuario o el correo ya se encuentren en la base de datos o en caso de que las contraseñas introducidas no sean iguales, nos mostrará el error de la figura 4 y 5 respectivamente. Una vez que se haya registrado volverá a la página anterior.

Si el usuario ha olvidado su contraseña clicará en “He olvidado mi contraseña”, con lo que nos dirigiremos a la figura 2. Aquí el usuario introducirá su correo y se auto-enviará la nueva contraseña donde desee. En el caso de no existir ese usuario se mostrará un error como en la figura 6.

Por último en esta pantalla (1) podremos realizar el inicio de sesión introduciendo el correo del usuario y la contraseña. En el caso de que alguno de estos valores esté mal introducido, mostrará el error de la pantalla 7. En caso de introducirlo correctamente y ser la primera vez que accedemos pasaremos a la pantalla 8.

En esta pantalla (en la 8), podremos ver todas las opciones pulsando en el botón de la esquina superior derecha, con lo que visualizaremos lo que se representa en la figura 9.

A partir de aquí, podremos acceder al resto del funcionamiento de la aplicación. Si pulsamos sobre “Ver perfil”, nos dirigiremos a la pantalla 10. Si pulsamos sobre el botón rotulado como “Editar”, pasaremos a la pantalla 11. En el caso de que queramos cambiar la contraseña y las dos no sean iguales se nos mostrará el error de la pantalla 12.

Por otro lado, el usuario podrá introducir un coche. Para ello, tendrá que pulsar, en las opciones, sobre “Introducir coche”, con lo que estaremos a continuación en la figura 13. En el caso de que no introduzcamos ningún parámetro, veremos el error de la figura 14. Si anteriormente ya se ha introducido un coche con ese nombre, aparecerá el error de la figura 15.

Además, el usuario podrá introducir trayectos. Si, por un lado, no está encendido el GPS, se mostrará un error como el de la figura 16. En el caso de que el usuario no haya introducido un coche pero sí tenga el GPS, aparecerá el error de la figura 17. Si por el contrario, tenemos tanto el GPS activado como un coche introducido, pasaremos a la figura 18. Pondremos el nombre del trayecto y seleccionaremos uno de los coches que tengamos introducidos. Tras esto, nos aparecerá la pantalla 19 durante todo el trayecto, cambiando la velocidad, la vía en la que está y el tiempo que se lleva de viaje. Si durante el viaje se

desconecta el GPS, la aplicación no introducirá ninguna ubicación. Una vez acabado el trayecto, se volverá a la pantalla principal (rotulada con el número 21).

La siguiente funcionalidad es ver la clasificación, la cual se muestra en la pantalla 20. Aquí podremos observar la posición del usuario, así como donde se encuentran el resto. En este ejemplo solo tenemos un usuario por lo que solo se nos mostrará a dicho usuario.

En el caso en el que ya tengamos algún trayecto realizado, en la pantalla principal, en vez de mostrar la figura 8, mostrará la 21. Si pulsamos sobre alguno de ellos podremos comparar distintos trayectos (lo explicaremos con la siguiente pantalla debido a que el funcionamiento es igual).

Para ver los trayectos, podemos pulsar sobre “Ver trayectos”. Esto nos llevará a una pantalla como la que aparece en la figura 22. Seleccionaremos el coche para el cual queremos ver sus trayectos, y nos aparecerá una pantalla como la 23 si hay algún trayecto, o como la 24 si ese coche aún no tiene trayectos. Si en la pantalla 23, pulsamos sobre algún trayecto, pasando así a la figura 25, nos aparecerán todos los trayectos con ese nombre (en este caso se han realizado dos veces el mismo trayecto en el mismo día). Podremos seleccionar hasta 3 trayectos, con los que nos llevará a la figura 26 (en este caso los dos trayectos). En ella se nos mostrarán las gráficas junto con su leyenda. Si realizamos un deslizamiento vertical en la pantalla, podremos ver cuales son las multas más relevantes de cada uno de los trayectos, para que así el usuario sepa cuáles podrían ser sus multas si pasasen por un radar.

### **5.5.2 Comparación con la maqueta**

Para la realización de la aplicación, se ha intentado seguir la maqueta inicial de la manera más fiel posible al diseño original. Como se puede observar en las pantallas del apartado anterior y en las maquetas disponibles en el Anexo A, las pantallas de inicio de sesión, registro, recuerdo de la contraseña, la pantalla principal, la introducción de un coche y la clasificación son muy similares. Sin embargo, las opciones no son iguales en todas ellas debido a que en la maqueta se planteó hacerlas de manera dinámica, para poder ver los trayectos de los coches desde la pantalla principal. Cuando se realizó la implementación, sin embargo, se consideró que era demasiado complejo, por lo que se realizó otra pantalla para seleccionar el coche y debajo irán apareciendo los trayectos (figura 22, 23 y 24).

En la introducción de la ruta, en la maqueta se pensó introducir el origen y el destino. Al realizar la implementación, simplemente para obtener el origen y el destino se recuperarán todos los tramos guardados y al ordenarlos por fecha el primero será el inicio y la última el fin.

Por último, se ha decidido modificar la pantalla que aparecerá durante el trayecto para introducir más información (como el tiempo transcurrido o la vía por la que se circula) en lugar de solo la velocidad.

## **5.6 Lectura de las gráficas**

La aplicación desarrollada, como se ha venido comentando a lo largo del presente documento, recogerá los datos de las ubicaciones por las que pasa el usuario a lo largo de un trayecto. De esta forma, se generará, a modo de resultado o resumen del trayecto, una gráfica que representará la velocidad a la que se ha movido el usuario en función del tiempo empleado en realizar dicho trayecto.

### **5.6.1 Estudio de las gráficas**

El usuario deberá realizar un estudio sobre las gráficas obtenidas para poder ser consciente de su conducción. Podrán aparecer hasta cuatro líneas, dependiendo de los trayectos que el usuario haya seleccionado. El número máximo de trayectos que se pueden comparar es tres (tendrán los colores azul, verde y amarillo).

Por otro lado, tendremos la velocidad máxima de la vía representada siempre en rojo, para que el usuario tenga una referencia de cómo de deprisa ha ido con respecto al máximo legal permitido, es decir, en caso de que la línea del color del trayecto esté por encima de la línea roja, significará que el usuario está cometiendo una infracción y, por el contrario, si se encuentra por debajo indicará que el usuario está respetando la velocidad máxima establecida.

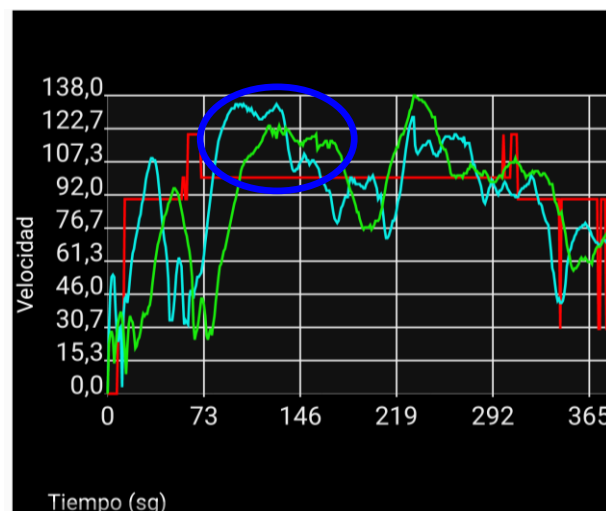
Además, podemos observar si el usuario ha realizado frenadas bruscas, debido a que en la gráfica se observará una línea que baja desde la velocidad en la que nos encontrábamos inicialmente hasta una velocidad mucho menor. Con esto, el usuario debería ser consciente de varias cosas: en el caso de que haya encontrado un atasco, debería adelantarse a los acontecimientos y siempre que se pueda ir frenando de manera más progresiva. En el caso que el usuario haya frenado por voluntad propia, debería intentar cambiar ese hábito para evitar futuros accidentes con el resto de vehículos de la vía.

Podemos encontrar una forma muy usual en las gráficas llamada “dientes de sierra”, esto significa que el usuario no lleva una velocidad constante durante el camino, sino que la velocidad oscila de manera brusca y repetitiva. Para una conducción eficiente este factor debería ser uno de los principales que el conductor debería evitar.

Para finalizar, en las gráficas, hay puntos en los que la velocidad máxima de repente es muy baja en comparación con la velocidad que se indicaba como máxima pocos metros más atrás (es decir, se observa una bajada repentina en la línea que indica la velocidad máxima de la vía, seguida de una subida de la misma magnitud). Esto es debido a que el usuario ha pasado por un túnel o bajo un puente y el GPS indica que se encuentra en una calle diferente a la que en realidad se encuentra, es decir, una carretera que pase por encima o una calle en territorio urbano, cuya velocidad es distinta (usualmente, es bastante menor).

### 5.6.2 Ejemplo de lectura

En este apartado, se enseñará un ejemplo, para poder mostrar los elementos más representativos de la gráfica (Figura 5-3).



**Figura 5-3: Gráfica de trayecto**

Antes de comenzar a analizar la gráfica hay que destacar que no representa un trayecto completo. En la imagen se muestra con una línea roja la velocidad máxima del trayecto y en azul y verde dos trayectos realizados. Como podemos comprobar, el usuario que ha realizado este trayecto, ha cometido infracciones, debido a que ha superado la velocidad máxima.

Además, se observa que en ciertas partes (lo rodeado de azul oscuro) ha tenido cambios de velocidad y no ha llevado una velocidad constante durante ese periodo de tiempo. Esto es a lo que nos referíamos unas líneas más arriba con los “dientes de sierra”.

Por otro lado, al final de la gráfica, tenemos una bajada de la línea que representa la velocidad máxima, la cual vuelve a subir en el segundo siguiente, tal y como se ha dicho antes, esto posiblemente se debe a que el usuario ha pasado por un lugar donde el GPS no ha reconocido correctamente la calle debido a que ha pasado por debajo de un puente y encima hay otra vía o una calle de una ciudad en la cual se debe circular más lento.

En esta gráfica, se puede observar que el usuario ha realizado prácticamente las mismas interacciones durante ambos trayectos pero con segundos de diferencia. Después de hablar con el conductor, confirmó que no había sido exactamente el mismo trayecto si no que cogió un par de rotondas más antes de realizar el mismo que el anterior, lo cual explica dicho desfase temporal.



## 6 Pruebas y resultados

---

### 6.1 Introducción

Las pruebas se han ido realizando a la vez que se ha ido desarrollando la aplicación, debido a que ciertas partes de código dependen de otras anteriormente implementadas, como ya se apuntó en la sección en el que se explicó la metodología que se iba a seguir. Las pruebas son un proceso de análisis del sistema construido, y su objetivo es comparar el comportamiento existente del sistema con el requerido.

Lo que se busca es encontrar el número máximo de errores posibles para entregar al usuario un sistema sin ellos en la medida de lo posible. Estas pruebas han sido realizadas con el dispositivo móvil *LG G4*.

En este proyecto se han realizado pruebas unitarias, de usabilidad, de validación y de verificación. Todas ellas las explicaremos en los siguientes apartados.

### 6.2 Pruebas unitarias

Estas pruebas son usadas para comprobar el correcto funcionamiento de una unidad de código. Estas pruebas han sido realizadas sobre los módulos de inicio de sesión, registro del usuario, introducción del coche e introducción de una ruta. No se ha probado la realización de un trayecto, debido a que se necesita que el sistema GPS registre puntos de localización de cada momento para probarlo, y no se disponía de un vehículo para poder probar cada uno de los puntos de dicha funcionalidad. Este punto se llevará a cabo en las pruebas de verificación.

### 6.3 Pruebas de usabilidad

La prueba consiste en que los usuarios interactúen con la aplicación, de tal manera que se pueda testear toda la funcionalidad de la aplicación. Una vez la aplicación es utilizada por todos los usuarios, podremos medir la usabilidad empleando la escala SUS (System Usability Scale). Los usuarios que hayan probado la aplicación, rellenan un cuestionario de diez ítems, cuyo resultado proporciona un porcentaje que indica en qué medida la aplicación es usable o no.

El cuestionario que se ha seguido se adjunta en el Anexo B. En él no sólo se incluyen las preguntas del cuestionario, sino que además contiene otras preguntas de control para poder argumentar el resultado y, por último, se pregunta por las ventajas, desventajas y aspectos a mejorar de la aplicación, para poder así añadir como trabajo futuro aquello que los usuarios más demanden si nos parece relevante para un mejor funcionamiento de la aplicación.

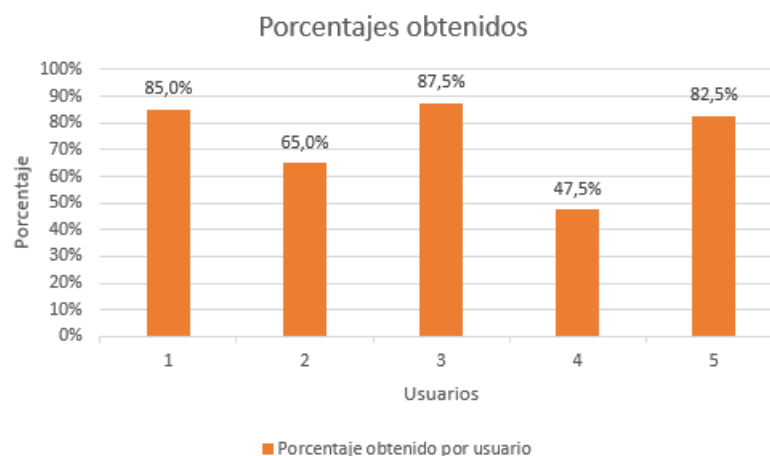
El cuestionario se compone de diez juicios de valor sobre diferentes aspectos de la aplicación, numerados del uno al diez. El usuario podrá calificar cada uno de ellos con una valoración entre uno y cinco, representando el uno el mayor grado de desacuerdo con la afirmación presentada, y cinco el mayor grado de acuerdo.

Para calcular la puntuación SUS, procederemos de la siguiente manera:

- Si la afirmación está etiquetada con un número impar, se restará uno a la puntuación otorgada por el usuario.
- Si, por el contrario, la afirmación tiene un número par, se restará cinco menos la valoración emitida por el usuario para dicha afirmación.

De esta manera, se obtendrán diez valores, uno para cada una de las afirmaciones. Acto seguido, se realizará la suma de estos valores y se multiplicará el resultado por 2.5, obteniendo una puntuación entre 0 y 100.

**Cinco usuarios han rellenado las preguntas**, habiendo utilizado todos y cada uno de ellos la aplicación en mayor o menor grado. Los resultados se pueden apreciar en la Figura 6-1.



**Figura 6-1: Gráfica de puntuación de los usuarios**

Tras estos resultados, realizamos la media de los porcentajes obtenidos para los distintos usuarios, y obtenemos que el sistema tiene un porcentaje de usabilidad de un 73,5%. SUS establece que si el sistema obtiene más de un 68% es usable, por lo que **podemos concluir que la aplicación desarrollada es usable, siendo este uno de los requisitos no funcionales que se planteó.**

Aunque el resultado indique que el sistema es usable, vemos que hay un usuario cuya puntuación está por debajo del 50%. Después de examinar sus resultados, se ha comprobado que el usuario ha indicado que no utiliza dispositivos móviles con frecuencia, lo cual hace que se expliquen de alguna manera las malas puntuaciones otorgadas a los diferentes puntos de la usabilidad del sistema. Este usuario considera poco intuitiva la aplicación desarrollada, pero muy probablemente considere poco intuitiva también la interacción con dispositivos móviles y aplicaciones diseñados para los mismos en general. Es probable que, para aprender a utilizar esta herramienta, esta persona requiera la ayuda de alguien que esté más familiarizado con este tipo de dispositivo.

Además, en el cuestionario se incluye un apartado para que el usuario defina cuales son los puntos positivos, los puntos negativos y aquellos aspectos que mejoraría de la aplicación.

Después de examinar todas las respuestas, lo más positivo que han detectado los usuarios ha sido:

- La comparación de viajes sucesivos conlleva a una mejora de la conducción, debido a que vas corrigiendo fallos, al tiempo que sigues viendo otros, lo cual te ayuda a mejorarlos.
- La aplicación lleva a los usuarios a un reto personal, en el cual desean no realizar ninguna incidencia, es decir, viajan a una velocidad menor.
- Como consecuencia de la anterior, los usuarios consumen menos combustible y van más relajados en la carretera.

Por otro lado, los aspectos negativos a destacar han sido:

- Los coches que se incluyen no todo el mundo los conoce, y además, faltan muchos coches comunes en España.
- Durante el trayecto, tener la visualización de un mapa para poder ver exactamente por dónde viaja el usuario.

Las sugerencias para mejorar la aplicación han sido:

- Enfatizar en mayor medida la faceta social de la aplicación, pudiendo compartir trayectos entre usuarios.
- Añadir gráficas con parámetros diferentes.
- Sería beneficioso que se guardase la sesión cuando el usuario quiera volver a entrar en la aplicación.

## **6.4 Prueba de validación**

Esta prueba consiste en evaluar un sistema o un componente del sistema para poder determinar si este satisface las condiciones impuestas al principio. Estas pruebas se han realizado a los casos de uso de introducir coche y añadir trayecto, ya que son los más representativos de la aplicación desarrollada. Estas pruebas se detallan en el Anexo C.

## **6.5 Pruebas de verificación**

Estas pruebas consisten en ver si el trabajo desarrollado es realmente lo que el usuario reclama. Para ello, se ha contado con un usuario, el cual ha probado durante 15 días la aplicación, para todos y cada uno de sus trayectos diarios.

El usuario, tras utilizarla, ha informado de su perfecto funcionamiento, además de la exactitud con la que mide la velocidad y la ubicación actual. La aplicación registra correctamente los trayectos realizados por el usuario, sin presentar cierres forzosos inesperados, reinicios ni ningún otro comportamiento errático que enturbie la experiencia de uso.

El usuario, además, ha expresado estar muy conforme con la aplicación, y ha manifestado que le encantaría seguir utilizándola y ver los avances que se vayan realizando si se sigue con el proyecto.

También ha expuesto el reto que le supuso disminuir la velocidad a la que indicaba la vía. Después de varios trayectos, ha conseguido no tener ninguna sanción. Ahora está intentando llevar una velocidad lo más constante posible para poder ver como sus gráficas mejoran.

Tras verificar y analizar el feedback de este usuario, podemos concluir que la aplicación responde correctamente a las pruebas de verificación, puesto que la funcionalidad final presentada al usuario final corresponde con lo que este espera de ella, habida cuenta de los requisitos iniciales definidos.

## 7 Conclusiones y trabajo futuro

---

### 7.1 Conclusiones

Una vez finalizado el proyecto, es importante mencionar las diferentes conclusiones a las que se ha llegado tras la realización de este proyecto. Estas conclusiones se pueden dividir en conclusiones de los objetivos del proyecto y conclusiones personales.

#### 7.1.1 Conclusiones de objetivos

En la introducción, se definieron todos los requisitos que se esperaban realizar en el Trabajo Fin de Grado. Para llevar a cabo estos propósitos, se ha desarrollado una aplicación móvil para que el usuario pudiese realizar trayectos, y así valorar su conducción. De tal forma que podrá ver cuáles son sus errores más comunes y poder mejorarlos poco a poco.

Tras desarrollar la aplicación pasando por las diferentes fases de diseño, implementación análisis y pruebas, podemos concluir diciendo que se han cumplido todos los requisitos. Pero no solo se han cubierto los requisitos, sino que, también, el usuario que ha más ha probado la aplicación establece que se ha encontrado muy cómodo con ella, y que le gustaría seguir utilizándola y ver los avances que esta podría llegar a tener. Incluso, se ha conseguido que dicho usuario reduzca la velocidad para poder demostrarse a sí mismo que puede respetar las normas de la vía y que puede conducir mucho mejor de lo que lo hace en este momento. Motivar a los usuarios a modificar sus hábitos de conducción y conseguir que lleven a cabo otros más eficientes y beneficiosos para todos era una de las principales motivaciones de este proyecto, por lo que este punto cobra una especial relevancia a la hora de cumplir con el objetivo descrito al principio del presente documento.

Por último, gracias a las pruebas, podemos concluir que se ha generado una aplicación usable, con un funcionamiento fácil e intuitivo para el usuario, que motiva al usuario a seguir utilizándola gracias al refuerzo positivo que supone superar los retos personales que la aplicación inspira.

### **7.1.2 Conclusiones personales**

Este proyecto ha supuesto un reto personal, debido a que ha sido un proyecto grande que he debido realizar sola. Además, con él he aprendido a planificarme y a dedicarle todo el tiempo posible para poder así, en los márgenes establecidos, llevar a cabo todos los requisitos.

También he aprendido que al principio del proyecto se sabe lo que se quiere hacer, pero cuando llegas a la fase de implementación te encuentras problemas que no se habían previsto, y se debe encontrar una solución lo más rápido posible para poder seguir adelante, o plantear alternativas que nos sirvan de igual forma que las ideas previas.

Dicho esto, estoy orgullosa de la aplicación resultante, aunque para ello haya tenido que aplicar mucho esfuerzo y muchas horas, ya que al final todo esfuerzo tiene su recompensa.

### **7.2 Trabajo futuro**

De cara al futuro se ha plantado introducir los siguientes puntos:

- Por un lado, añadir más gráficas de parámetros diferentes, para que el usuario pueda tener más formas de juzgar su conducción, así como comparar los mejores trayectos del mes, o incluso comparar los mismos trayectos con otros usuarios.
- Añadir mapas donde se pinten los puntos GPS, para que el usuario pueda identificar el origen y el destino. Además, señalar en rojo (en el mapa) aquellas zonas donde el usuario haya cometido una infracción y de verde donde haya respetado las normas de velocidad.
- Se podría añadir un módulo de revisiones, es decir, el usuario podría introducir cuando quiera que se le avise de cuando está próxima a vencer la ITV en vigor de alguno de los vehículos que utiliza y tiene registrados en la aplicación, o de cuándo será necesario el próximo cambio del aceite y el sistema, llegado ese momento, le avisará mediante una notificación.
- También se puede advertir al usuario de las condiciones meteorológicas, es decir, si se espera que en cierta zona nieve, por ejemplo, se le debe avisar al usuario de que lleve cadenas antes de realizar el viaje.
- Además, a lo largo del viaje, se puede avisar al usuario de cuándo debe descansar y presentarle la ubicación de puntos cercanos a su localización para que pueda realizar dicha acción.
- Otro aspecto importante de la conducción también podría ser controlar si el usuario se está cambiando constantemente de carril, gracias a los datos acerca de los movimientos laterales que registra el dispositivo móvil.

# Referencias

---

- [1]Ministerio del Interior, “Conducción eficiente.” [Online]. Available: [http://www.dgt.es/PEVI/documentos/catalogo\\_recursos/didacticos/did\\_adultas/Conduccion\\_eficiente.pdf](http://www.dgt.es/PEVI/documentos/catalogo_recursos/didacticos/did_adultas/Conduccion_eficiente.pdf). [Accessed: 07-Jun-2018].
- [2]E. Martínez, “Drive Smart, la mejor forma de conducir,” 2015. [Online]. Available: <http://www.abc.es/tecnologia/moviles-aplicaciones/20150317/drive-smart-mejor-forma-201503161548.html>. [Accessed: 08-Jun-2018].
- [3]“DriveSmart.” [Online]. Available: <http://drive-smart.com/es/>. [Accessed: 08-Aug-2018].
- [4]“OpenStreetMap.” [Online]. Available: <https://www.openstreetmap.org/>. [Accessed: 09-Jun-2018].
- [5]“Waze,” 2006. [Online]. Available: <https://www.waze.com/es/>. [Accessed: 09-Jun-2018].
- [6]“Velociraptor.” [Online]. Available: <https://play.google.com/store/apps/details?id=com.pluscubed.velociraptor&hl=es>. [Accessed: 20-Jun-2018].
- [7]J. Alviz, “iOS contra Android ¿quién gana la batalla?,” 2013. [Online]. Available: <https://clipset.20minutos.es/ios-contra-android-quien-gana-la-batalla/>. [Accessed: 08-Jun-2018].
- [8]“Metodología RUP.” [Online]. Available: <https://procesosdesoftware.wikispaces.com/METODOLOGIA+RUP>. [Accessed: 06-Jun-2018].
- [9]“Metodología Ágil Proceso Unificado de Rational RUP.” [Online]. Available: <http://audiemangt.blogspot.com/2010/05/metodologia-agil-proceso-unificado-de.html>. [Accessed: 19-Jun-2018].
- [10]“Metodología RUP.” [Online]. Available: <http://lorentarache8.blogspot.com/p/metodologia-rup.html>. [Accessed: 19-Jun-2018].
- [11]“Descargar Android Studio.” [Online]. Available: <https://developer.android.com/studio/>. [Accessed: 07-Jun-2018].
- [12]“Crear un proyecto.” [Online]. Available: <https://developer.android.com/studio/projects/create-project?hl=es-419>. [Accessed: 07-Jun-2018].
- [13]“Proyectos Recientes.” [Online]. Available: <https://developer.android.com/studio/projects/?hl=es-419>. [Accessed: 07-Jun-2018].
- [14]“Usando Git y Bitbucket” 2012. [Online]. Available: <http://www.memoriasdeunaprendiz.com/programacion/git-gitbucket/>. [Accessed: 07-Jun-2018].
- [15]W3School, “Extensible Markup Language (XML).” [Online]. Available: <https://www.w3schools.com/xml/>. [Accessed: 07-Jun-2018].
- [16]“SQLite Tutorial.” [Online]. Available: <https://www.tutorialspoint.com/sqlite/index.htm>. [Accessed: 07-Jun-2018].
- [17]W3School, “Structured Query Language (SQL).” [Online]. Available: <https://www.w3schools.com/sql/>. [Accessed: 07-Jun-2018].
- [18]“Android.” [Online]. Available: <https://www.android.com/>. [Accessed: 08-Jun-2018].
- [19]Arthur Pointeau, “Casos de uso,” 2018. [Online]. Available: <https://openclassrooms.com/courses/planea-tu-proyecto-con-uml-1/diagramas-de-casos-de-uso>. [Accessed: 18-Jun-2018].

- [20] F. Berzal, “Relaciones entre clases: Diagramas de clases UML.” [Online]. Available: <http://elvex.ugr.es/decsai/java/pdf/3C-Relaciones.pdf>. [Accessed: 07-Jun-2018].
- [21] J. M. Ramez Elmasri;Shamkant B.Navathe;Diaz, *Fundamentos de sistemas de bases de datos*, Pearson S. 2007.
- [22] “Data on Cars used for Testing Fuel Economy.” [Online]. Available: <https://www.epa.gov/compliance-and-fuel-economy-data/data-cars-used-testing-fuel-economy>. [Accessed: 18-Jun-2018].
- [23] D. G. de Tráfico, “Cuadro para excesos de velocidad.” [Online]. Available: [https://sede.dgt.gob.es/Galerias/tramites-y-multas/alguna-multa/consulta-de-sanciones-por-exceso-velocidad/cuadro\\_velocidad.pdf](https://sede.dgt.gob.es/Galerias/tramites-y-multas/alguna-multa/consulta-de-sanciones-por-exceso-velocidad/cuadro_velocidad.pdf). [Accessed: 20-Jun-2018].
- [24] “Modelo Vista Controlador.” [Online]. Available: <https://codigofacilito.com/articulos/mvc-model-view-controller-explicado>. [Accessed: 08-Jun-2018].
- [25] “Conectar Java a una base de datos,” 2016. [Online]. Available: <https://www.discoduroderoer.es/conectar-java-a-una-base-de-datos-sqlite/>. [Accessed: 27-Feb-2018].
- [26] Android Developers, “Saved data using SQLite.” [Online]. Available: <https://developer.android.com/training/data-storage/sqlite>. [Accessed: 12-Jun-2018]
- [27] “Location strategies.” [Online]. Available: <https://developer.android.com/guide/topics/location/strategies>. [Accessed: 12-Jun-2018]
- [28] “Limites de velocidad.” [Online]. Available: <https://developers.google.com/maps/documentation/roads/speed-limits?hl=es-419>. [Accessed: 11-Jun-2018].
- [29] “Autovías y Autopistas de España.” [Online]. Available: [https://es.wikipedia.org/wiki/Anexo:Autopistas\\_y\\_autovías\\_de\\_España](https://es.wikipedia.org/wiki/Anexo:Autopistas_y_autovías_de_España). [Accessed: 11-Jun-2018].
- [30] “Carreteras nacionales de España.” [Online]. Available: [https://es.wikipedia.org/wiki/Anexo:Carreteras\\_Nacionales\\_de\\_España](https://es.wikipedia.org/wiki/Anexo:Carreteras_Nacionales_de_España). [Accessed: 11-Jun-2018].
- [31] “Carreteras Comarcales de España.” [Online]. Available: [https://es.wikipedia.org/wiki/Anexo:Carreteras\\_Comarcales\\_de\\_España](https://es.wikipedia.org/wiki/Anexo:Carreteras_Comarcales_de_España). [Accessed: 11-Jun-2018].
- [32] C. E. Land, “Geocoder.” [Online]. Available: <https://developer.android.com/reference/android/location/Geocoder>. [Accessed: 11-Jun-2018].



## Glosario

---

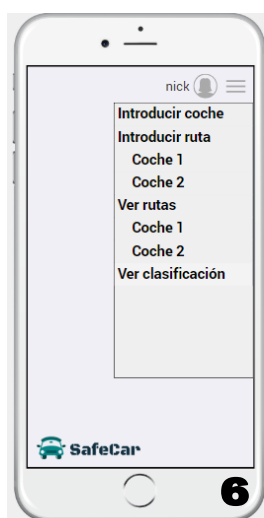
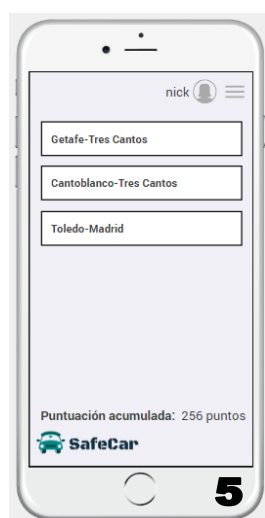
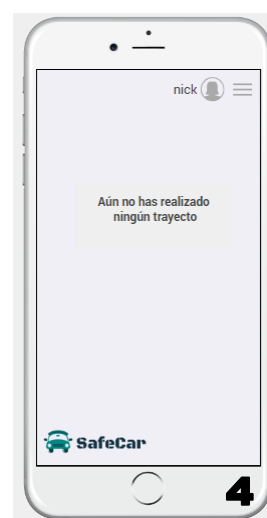
API	Application Programming Interface
SDK	Software Development Kit
GPS	Global Positioning System
RUP	Rational Unified Process
ITV	Inspección técnica de vehículos
SUS	Escala de Usabilidad de Sistema (System Usability Scale)

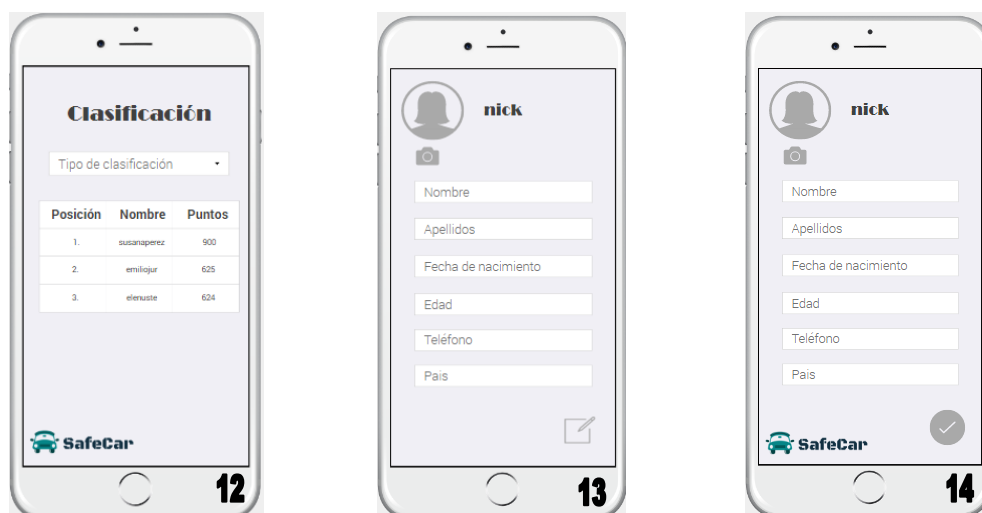


## Anexos

### A Maqueta

En este anexo, se detallará la maqueta realizada para la aplicación. De tal manera que mostraremos la conexión que existe entre todas ellas.





**Figura A-1: Maqueta de la aplicación**

- Pantalla de inicio de sesión.** En esta pantalla, el usuario deberá introducir su usuario y su contraseña si este está registrado para poder pasar a la pantalla 4 o 5. Si se ha realizado algún trayecto pasaremos a la pantalla 5, en caso contrario, a la 4. Si el usuario aún no se ha registrado, pasaremos a la pantalla 3. En el caso de olvidar la contraseña, iremos a la pantalla 2.
- Pantalla de recuperación de contraseña.** En esta pantalla, el usuario deberá introducir su correo para obtener una nueva contraseña, la cual le será enviada donde el usuario elija.
- Pantalla de registro.** Esta pantalla será la primera que deberá completar el usuario para poder utilizar la aplicación. Tras registrarse iremos a la pantalla 4.
- Pantalla de inicio.** Esta pantalla, junto a la 5, será el inicio, es decir, siempre veremos los últimos trayectos realizados; y en caso de no tener, aparecerá vacío. En esta pantalla de inicio podremos ver las opciones que tendrá la aplicación, pasando así a la pantalla 6. Además, se podrá acceder al perfil, es decir, la pantalla 13.
- Pantalla de inicio 2.** Igual que la anterior, con la diferencia de que aquí tendremos los últimos trayectos realizados.
- Pantalla de opciones.** Esta pantalla será la más importante de la aplicación, ya que nos lleva a toda la funcionalidad de ésta. Por un lado, si queremos introducir un coche, nos llevará a pantalla 7. Si queremos, por el contrario, introducir un trayecto para un coche determinado, nos llevará a la pantalla 8. Podremos también ver los trayectos o rutas realizadas, llevándonos así a la pantalla 9. Para ver la clasificación, iremos a la pantalla 12. Podremos ir también al perfil, mediante el icono situado al lado de las opciones.

7. **Pantalla de introducción de un coche.** En esta pantalla introduciremos un coche para después realizar rutas. Para ello, introduciremos los datos tales como la marca y el modelo.
8. **Pantalla de introducción de una ruta.** Al introducir una ruta deberemos introducir ciertos campos, una vez introducidos, podremos empezar el trayecto, pasando así a la pantalla 10.
9. **Pantalla de trayectos.** Podremos ver las rutas que hemos realizado con un coche. A esta pantalla se accede desde la pantalla 6. Tendremos la opción de pulsar sobre cada uno de los trayectos para ver como lo hemos realizado, pasando así a la pantalla 11.
10. **Pantalla de viaje.** Se indicará la velocidad a la que va el usuario durante el trayecto.
11. **Pantalla análisis de un trayecto.** En esta pantalla, se mostrará cómo hemos realizado un trayecto. Desde esta pantalla podremos ver la progresión del trayecto seleccionado.
12. **Pantalla de clasificación.** A esta pantalla accederemos a través de la pantalla principal (6), y en ella podremos consultar nuestro puesto en la clasificación.
13. **Pantalla perfil de usuario.** En esta pantalla podremos modificar algunos datos de usuario, pasando así a la pantalla 14.
14. **Pantalla guardar perfil de usuario.** Podremos guardar los cambios realizados, volviendo así a la pantalla 13.



## B Plantilla del Sistema de Usabilidad (SUS)

**Cuestiones generales:** Para cada una de las siguientes cuestiones marca la casilla correspondiente.

¿Eres usuario habitual de dispositivos móviles (teléfonos o tablets)? .....	Si	<input type="checkbox"/>	No	<input type="checkbox"/>
¿Eres conductor habitual? .....	Si	<input type="checkbox"/>	No	<input type="checkbox"/>
¿Sabes en qué consiste la conducción eficiente? .....	Si	<input type="checkbox"/>	No	<input type="checkbox"/>

**Instrucciones:** Para las siguientes afirmaciones, marca la casilla que mejor describa tus reacciones a la herramienta

	totalmente de acuerdo →					
	← totalmente en desacuerdo					
Creo que me gustaría usar esta herramienta con frecuencia .....	<table><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr></table>	1	2	3	4	5
1	2	3	4	5		
Encontré esta herramienta innecesariamente compleja .....	<table><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr></table>	1	2	3	4	5
1	2	3	4	5		
Creo que la herramienta es fácil de usar .....	<table><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr></table>	1	2	3	4	5
1	2	3	4	5		
Creo que necesitaría ayuda para poder usar esta herramienta .....	<table><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr></table>	1	2	3	4	5
1	2	3	4	5		
He encontrado que las diversas funciones de esta herramienta estaban bien integradas .....	<table><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr></table>	1	2	3	4	5
1	2	3	4	5		
Creo que hay demasiadas funciones inconsistentes en esta herramienta .....	<table><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr></table>	1	2	3	4	5
1	2	3	4	5		
Creo que la mayoría de la gente puede aprender a usar esta herramienta muy rápidamente .	<table><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr></table>	1	2	3	4	5
1	2	3	4	5		
He encontrado esta herramienta muy engorrosa/incómoda de usar .....	<table><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr></table>	1	2	3	4	5
1	2	3	4	5		
Me sentí muy seguro de lo que hacía al usar esta herramienta .....	<table><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr></table>	1	2	3	4	5
1	2	3	4	5		
Tengo que aprender un montón de cosas antes de poder usar esta herramienta .....	<table><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr></table>	1	2	3	4	5
1	2	3	4	5		

Por favor, indica tres aspectos positivos que quieras resaltar sobre la app:

---

---

---

Por favor, indica tres aspectos negativos de la app:

---

---

---

¿Tienes alguna sugerencia de mejora?:

---





## ***C Pruebas de validación***

### **C.1 Prueba para el caso de uso “Introducir coche”**

#### **Precondiciones:**

- El usuario se encuentra registrado en la aplicación y ha iniciado sesión.
- Se encuentra por tanto en la pantalla principal.

#### **Escenario:**

1. El usuario pulsa sobre el botón de opciones situado en la esquina superior derecha.
2. El sistema muestra las opciones disponibles: “Perfil”, “Introducir coche”, “Introducir trayecto”, “Clasificación” y “Ver trayectos”.
3. El usuario selecciona la segunda opción de la lista: “Introducir coche”.
4. El sistema muestra la pantalla de “Introducir coche”.
5. El usuario rellena los datos del coche: nombre, año, marca, modelo y tipo (automático, semi-automático o manual). A continuación pulsa sobre “Guardar”.
6. El sistema guarda los datos introducidos por el usuario y vuelve a la pantalla principal.

#### **Flujo alternativo 1:**

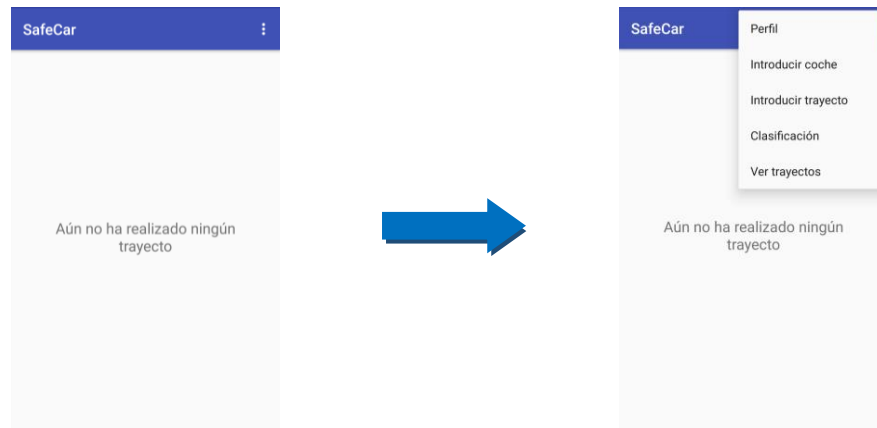
- 5.1. El usuario no introduce todos los campos obligatorios sobre las características del coche.
- 6.1. El sistema informa al usuario de que debe rellenar todos los campos, y queda a la espera de nueva interacción por parte del usuario.

#### **Flujo alternativo 2:**

- 5.2. El usuario introduce un nombre que ya ha sido utilizado anteriormente para otro vehículo.
- 6.2. El sistema informa al usuario de que ya existe ese nombre y debe introducir uno distinto.

## Resultado real de la prueba:

1. El usuario pulsa sobre el botón de opciones y 2. El sistema muestra las opciones disponibles:



**Figura C-2: Transición a las opciones de la aplicación**

3. El usuario pulsa sobre “Introducir coche”, y 4. El sistema muestra la pantalla de “Introducir coche”:



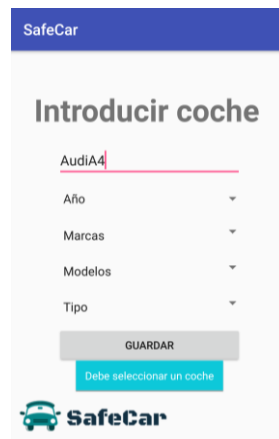
**Figura C-3: Transición para introducir un coche**

5. El usuario introduce el nombre y los datos del vehículo y pulsa sobre guardar.



**Figura C-4: Transición a la pantalla principal con trayectos**

Aquí podemos tener varios flujos distintos: en caso de que los datos introducidos sean correctos y no exista ya otro coche con ese nombre, se creará el trayecto satisfactoriamente y se volverá a la pantalla principal, tal y como muestran las capturas sobre estas líneas. Si por, el contrario, falta algún dato por rellenar, ocurrirá lo siguiente:



**Figura C-5: Introducir un coche**

Si lo que ocurre es que se está intentando introducir el coche con el mismo nombre de otro que ya existe en la base de datos de la aplicación, veremos lo siguiente:



**Figura C-6: Error al introducir un coche**

Como vemos, el funcionamiento de la aplicación desarrollada, tal y como muestran las capturas, corresponde con el que se espera para este caso de uso, por lo que podemos considerar el resultado de la prueba de validación como satisfactorio.

## **C.2 Prueba para el caso de uso “Introducir trayecto”**

### **Precondiciones:**

- El usuario se encuentra registrado en la aplicación y ha iniciado sesión. Se encuentra por tanto en la pantalla principal.
- Además, se debe haber introducido un coche previamente en la aplicación.
- El GPS se encuentra activado en el dispositivo en el que se está ejecutando la aplicación.

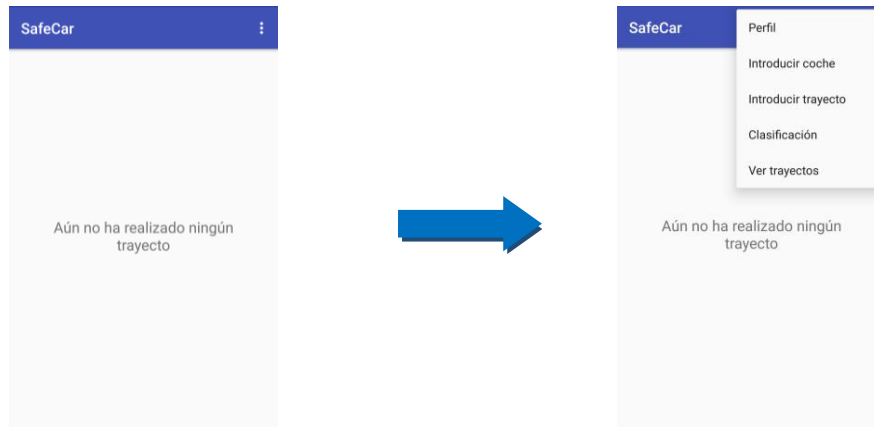
### **Escenario:**

1. El usuario pulsa sobre el botón de opciones situado en la esquina superior derecha.
2. El sistema muestra las opciones disponibles: “Perfil”, “Introducir coche”, “Introducir trayecto”, “Clasificación” y “Ver trayectos”.
3. El usuario selecciona la segunda opción de la lista: “Introducir trayecto”.
4. El sistema muestra la pantalla de “Introducir trayecto”.
5. El usuario rellena los datos del trayecto: nombre y coche que se utilizará. A continuación pulsa sobre “Comenzar”.

6. El sistema guardar los datos introducidos por el usuario y se dirige a la pantalla que se mostrará durante el viaje.

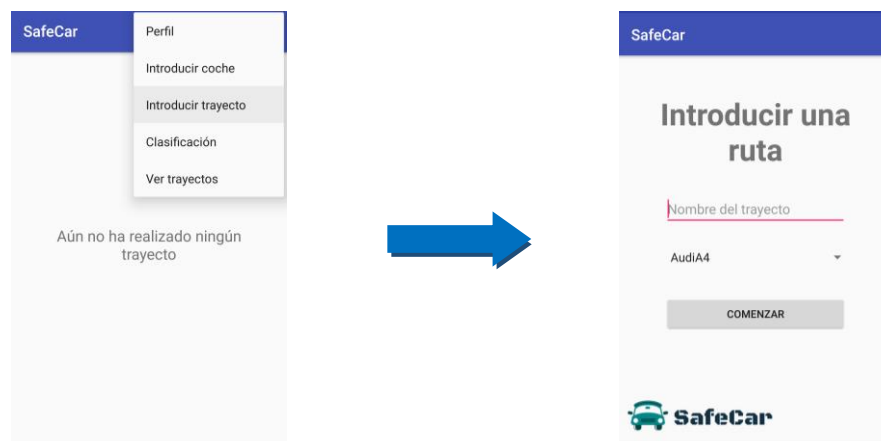
### **Resultado real de la prueba:**

1. El usuario pulsa sobre el botón de opciones y 2. El sistema muestra las opciones disponibles:



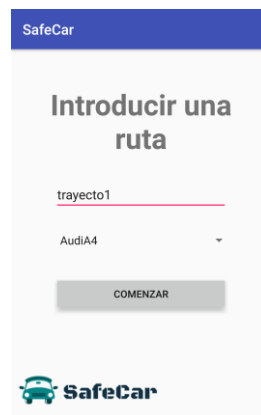
**Figura C-7: Transición a las opciones de la aplicación**

3. El usuario pulsa sobre “Introducir trayecto”, y 4. El sistema muestra la pantalla de “Introducir trayecto”:



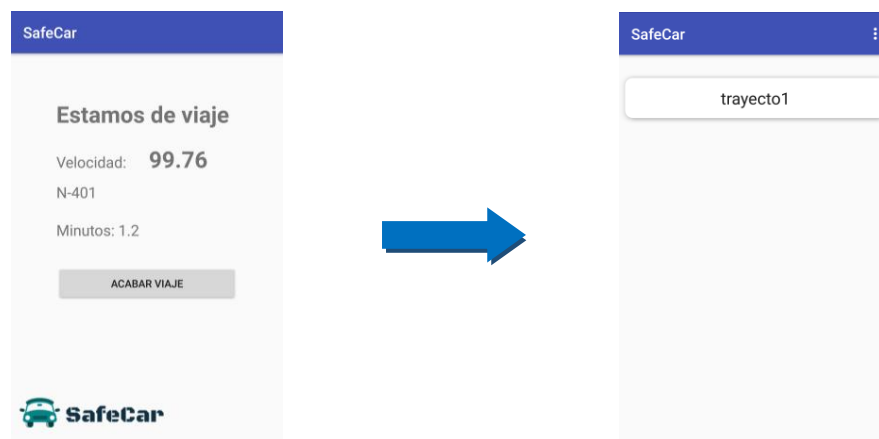
**Figura C-8: Transición para introducir un trayecto**

5. El usuario introduce el nombre y el vehículo y pulsa sobre “Comenzar”.

The image shows a mobile app interface for 'SafeCar'. At the top is a blue header with the text 'SafeCar'. Below the header, the title 'Introducir una ruta' is centered. There is a text input field containing 'trayecto1' with a red underline. Below the input field is a dropdown menu showing 'AudiA4'. At the bottom of the form is a grey button labeled 'COMENZAR'. The 'SafeCar' logo is at the very bottom.

**Figura C-9: Introducir una ruta**

6. El sistema nos dirige a la pantalla en la cual permaneceremos durante todo el trayecto. Una vez finalizado el trayecto, el usuario pulsará sobre el botón “Acabar viaje”. El sistema guardará el trayecto realizado en la base de datos y nos dirigirá de nuevo a la pantalla principal.



**Figura C-10: Transición de la finalización de un trayecto a la pantalla principal**

Como vemos, el funcionamiento de la aplicación desarrollada, tal y como muestran las capturas, corresponde con el que se espera para este caso de uso, por lo que podemos considerar el resultado de la prueba de validación como satisfactorio.